

Institut für Theoretische Physik
Fakultät für Physik und Geowissenschaften
Universität Leipzig

– diploma thesis –

Mass Condensation in Stochastic Transport Processes and Complex Networks

submitted by

Hannes Nagel

16.02.2010

tutors: Prof. Dr. Wolfhard Janke
Dr. Bartłomiej Waćław

referees: Prof. Dr. Wolfhard Janke
Prof. Dr. Ulrich Behn

Contents

List of Figures	III
List of Tables	V
Introduction and Overview	1
1. Statistical Physics	3
1.1. Thermal Equilibrium	3
1.2. Non-Equilibrium Processes	5
2. Numerical Methods	7
2.1. Exact Enumeration	7
2.2. Monte Carlo Methods	8
2.2.1. The Metropolis algorithm	9
2.2.2. Dynamic Metropolis updates	10
2.3. Error Analysis	11
2.4. Cluster Identification	13
3. Mass Condensation in Stochastic Transport	15
3.1. A Zero-Range Transport Process	16
3.1.1. The steady state of the ZRP	18
3.1.2. Condensation process	19
3.1.3. Condensation on inhomogeneous networks	19
3.2. Transport with Short Range Interaction and PFSS	22
3.2.1. The steady state of the PFSS process	22
3.2.2. Properties in the steady state	25
3.2.3. Dynamics of the condensation	28
3.3. Estimation methods	30
3.3.1. Properties of the steady state	30
3.3.2. Estimation of the condensation time scale	33
3.3.3. Remarks for the simulation on lattices and graphs	34
3.4. Results	35
3.4.1. Condensation on the periodic chain	35
3.4.2. Condensation on Regular and Disordered Lattices	45
4. Monte Carlo Generation of Complex Networks	49
4.1. Definitions in the Context of Graphs	50
4.2. Characteristics of Complex Graphs	51
4.3. Statistical Ensembles of Graphs	54

Contents

4.4. Monte Carlo Generation of Equilibrated Networks	57
4.4.1. Graph representation	58
4.4.2. Local updates and transition rates	59
4.4.3. The micro-canonical network ensemble	62
4.4.4. The canonical network ensemble	63
4.4.5. The grand-canonical network ensemble	66
4.5. Exemplary Graph Ensembles and Properties	68
4.6. Further Remarks	69
5. Conclusion and Outlook	73
A. Partition Function of the ZRP on scale-free Networks	75
B. Application of the Graphgen Library	77
Bibliography	81

List of Figures

2.1.	Description of the Hoshen–Kopelman algorithm	14
3.1.	Scheme of particle movement in the ZRP	16
3.2.	Phases of the ZRP	17
3.3.	Validation of time scale estimation with the ZRP	20
3.4.	ZRP on networks: Exact vs. estimate degree distribution	21
3.5.	Values of the hopping rate $u(m_i m_{i-1}, m_{i+1})$	25
3.6.	Phases of the PFSS process	26
3.7.	Site occupation number distribution	27
3.8.	Average envelope shape of the condensate	28
3.9.	Site occupation time series illustrating the condensation process	29
3.10.	Per site current in a system with two condensates	32
3.11.	First passage times of the heaviest condensate passing a mass threshold	36
3.12.	Time scale analysis of the heaviest condensate’s mass	37
3.13.	Time scale analysis of the average number of condensates in the PFSS system	38
3.14.	Time scale analysis of the average number of condensates in the PFSS system, $M = 3N$	39
3.15.	Distribution of first passage times for $n_{\text{textc}} = 1$	40
3.16.	Estimation of scaling exponent from distribution of first passage times	40
3.17.	Escape flow rate plotted versus condensate size.	42
3.18.	Shape of the rescaled current out of condensates	44
3.19.	Difference of currents between condensates in the coarsening regime	44
3.20.	Shape of a condensate on the two dimensional lattice	45
3.21.	Time scale estimation of condensation on a lattice	46
3.22.	Time scale estimation of condensation on regular graphs	47
4.1.	The possible labeled graphs with $N = 4, L = 2$	55
4.2.	Examples of different classes of graphs	58
4.3.	Example graph for discussion of representation schemes	59
4.4.	Overview of the fundamental local graph update mechanisms	60
4.5.	Details for the microcanonic update mechanism	62
4.6.	Details for the canonic update mechanism	63
4.7.	Details for the grandcanonic update mechanism	66
4.8.	Properties and sample of a graph ensemble mimicing the degree distribution Barabási–Albert networks	70
4.9.	Properties and sample of a graph ensemble tuned to favor disas- sortative networks	71

List of Figures

4.10. Properties and sample of a graph ensemble adjusted towards assortative networks	72
B.1. Minimal example using the graphgen library	78
B.2. Schematic view of module interaction in graphgen	79

List of Tables

3.1. Simulation effort for the different parameters	35
4.1. Adjacency list representation example	60
4.2. Mechanics of update construction in the generation of networks .	61

Introduction and Overview

Statistical physics is a field of physics which applies statistical methods to predict macroscopic properties of physical systems from their microscopic model of properties and interactions. For example, in solid state systems the properties in thermal equilibrium are modeled by interactions on the length scale of atoms and molecules. The size of such many-particle systems, is what enables these methods but prevents treatment with classical methods at the same time. A most typical application of statistical physics is the description of properties of Hamiltonian systems in thermal equilibrium with their environment. However, also non-Hamiltonian systems, without a properly defined energy, may be treated by these methods. For example, the stochastic transport processes discussed in this thesis represent such systems.

Stochastic transport describes movement of a conserved quantity, from one location in space to another in terms of classical stochastic processes. This quantity usually is a generic “mass”, such as molecules diffusing in a porous crystal, sand grains blown by the wind [1], people moving in traffic infrastructures, i.e. particles performing a stochastic motion on a structure. These processes may have a form where a current of the generic mass is driven through the system, in which case one speaks of driven diffusive systems. The drive therefore may hold such a system in a steady state out of equilibrium.

An interesting feature in driven diffusive systems is that, above a certain density, mass condensation may be observed. That is, a finite fraction of all masses accumulates at a certain site. The interpretation depends what the masses mean: Sand grains are piled up to form dunes [1] or vehicles cause traffic jams. Other interesting examples are structure formation by self-organization in crystal growth processes [2], or the dynamics of people (“masses”) surfing the world wide web.

Besides the diverse meanings of “mass” in these examples, the major difference is in the structures the processes take place on. Crystals grow differently on distinct lattice types and traffic jams are more likely on some structures than on others. Therefore it is an interesting question to ask, what the impact of structure to condensation is.

As for the effect of structure on processes, it is interesting to study the properties of complex networks. These are networks that exhibit several structure properties observed in real-world networks but not common to random graphs. This opens for example the possibility to study the robustness of specific types of networks to disappearance of nodes or the evolution of networks in interaction with a stochastic process on them [3].

In this thesis I will discuss the process of mass condensation in a specific stochastic transport model with the focus on the scaling of the condensation time. The condensation transition is first studied in the one dimensional model, using

numerical methods to estimate properties of the steady state and such describing the dynamics. From this basis the dynamics on the two dimensional square lattice is quickly evaluated. However, the step to study the condensation process on networks with respect to specific structure elements in complex networks is difficult. It is first necessary to create a means to allow the selective generation of networks with the desired structure properties. Then the condensation process may be studied starting with rewired regular lattices as structures.

This diploma thesis is organized as follows:

Chapter 1 gives a brief introduction into statistical physics. The focus lies on non-equilibrium processes and phase transitions.

Chapter 2 introduces the employed numerical methods that are used to simulate the dynamics of the transport model as well as its steady state. It also deals with the methods to determine the numerical errors of the obtained data and describes a generalization of the Hoshen–Kopelman cluster identification algorithm that works on connected graphs.

Chapter 3 first discusses a simple transport model without a ranged interaction that exhibits a condensation transition. On that basis, the studied stochastic transport model is described and properties in the steady state of the system are discussed. Then, the specific estimation methods used for the analysis of the condensation time scale and condensation process are layed out followed by a discussion o the obtained results. This is done in detail for the one dimensional system with asymmetric dynamics followed with an examination of the process on two dimensional lattices and regular graphs.

Chapter 4 gives a brief introduction to the statistical mechanics of complex networks up to the definition of statistical ensembles of graphs on the basis of Erdős–Renyi model of random graphs. Different basic ensembles are defined and a toolkit for the generation of complex networks out of these ensembles is presented.

Chapter 5 gives a short summary of the results and an outlook to further research on the topic.

1. Statistical Physics

Statistical physics is a framework providing formalisms and methods to deal with many particle systems employing statistical methods. These methods enable the derivation of properties of a macroscopic system from the microscopic interactions of its components. Each such macroscopic state corresponds to a set of microstates, i.e. complete descriptions of the state of each microscopic component, that give the same macroscopic properties.

An important concept basic to the used simulation methods is the Markov process. As well, stochastic mass transport processes, are also described by Markov process. A Markov process is a model for the random evolution of a memoryless system, i.e. the probability of the system to occupy any given state in the future depends only on its present state. A Markov process with a discrete state-space, as is the case during this thesis, is often referred to as a Markov chain. An illustrating example is to draw a number of marbles from an urn (containing different kinds of marbles) without replacement. The probabilities of different outcomes may be predicted at any time in the process, but depend only on the current partition of marbles that were drawn, never on the sequence they were drawn. The use of Markov processes enables the used simulating techniques and data analysis methods is discussed in the next chapter.

In this chapter I will briefly present the employed concepts in statistical physics. It is divided into the sections about systems in thermal equilibrium and non-equilibrium. For a more thorough and detailed introduction of statistical physics the reader may refer to statistical physics textbooks like [4, 5].

1.1. Thermal Equilibrium

A system is said to be in equilibrium with a heat bath, if there is no heat current from nor to the heat bath. A system in equilibrium is sufficiently described by its macroscopic properties. As the probability of the system to be in a certain microstate \vec{m} , given by the Boltzmann distribution

$$P(\vec{m}) = \frac{1}{Z} e^{-\beta H(\vec{m})}, \quad (1.1)$$

only depends on the macroscopic energy $H(\vec{m})$, all microstates with the same energy must be equiprobable. Therefore a statistical ensemble can be constructed with a uniform probability measure that contains all microstates of a certain energy. This statistical ensemble is called the micro-canonical ensemble. In it, the expectation value of an observable \mathcal{O} takes the form

$$\langle \mathcal{O} \rangle_E = \frac{1}{\Omega(E)} \sum_{\substack{\vec{m} \\ H(\vec{m})=E}} \mathcal{O}(\vec{m}), \quad (1.2)$$

1. Statistical Physics

where the normalization $\Omega(E)$ is the number of microstates with energy E , namely the density of states. In this picture, the ensemble is a collection of copies of a system, each occupying a microstate. The evolution of a system can then be seen as a walk stepping on these copies in a way defined by the dynamics of the system. If now the dynamics allow to reach any such copy from another one in finite time, the ergodic hypothesis states that the ensemble expectation (1.2) can be taken as the expectation of the system evolving in time. Indeed, the same fundamental reasoning is used to work with any statistical ensemble in statistical physics.

The energy of a real system that is in equilibrium with a heat bath is not fixed but fluctuates around its expectation value for the temperature of the heat bath. Therefore, a statistical ensemble is defined with a fixed temperature, containing any microstates of the system, that has a fixed size (i.e. number of particles). The Gibbs–measure then assigns the correct equilibrium probabilities to the microstates and the expectation of an observable \mathcal{O} becomes

$$\langle \mathcal{O} \rangle_\beta = \frac{1}{Z_N} \sum_{\vec{m}} \mathcal{O}(\vec{m}) e^{-\beta H(\vec{m})} = \frac{1}{Z_N} \sum_E \langle \mathcal{O} \rangle_E e^{-\beta H(\vec{m})} , \quad (1.3)$$

with the inverse temperature $\beta = 1/(k_B T)$ and the normalization Z , which is the canonical partition function

$$Z_N = \sum_{\vec{m}} e^{-\beta H(\vec{m})} = \sum_E \Omega(E) e^{-\beta E} . \quad (1.4)$$

The evolution of the probabilities of microstates of a system is described by the Master equation. For a system with discrete states it takes the form

$$\frac{dP_\alpha(t)}{dt} = \sum_\alpha (T_{\alpha\beta} P_\beta(t) - T_{\beta\alpha} P_\alpha(t)) , \quad (1.5)$$

with the transition matrix $T_{\alpha\beta}$. The elements $T_{\alpha\beta} = W(\alpha \rightarrow \beta)$ correspond to the transition rates of the system from a microstate α to a microstate β . For a system in equilibrium, the state probabilities are constant $\partial_t P_\alpha(t) = 0$, hence the left hand side becomes zero and the Master equation exhibits *balance*

$$\sum_\alpha T_{\alpha\beta} P_\beta = \sum_\alpha T_{\beta\alpha} P_\alpha . \quad (1.6)$$

That is, the probability current out of any configuration is equal to the probability current into this configuration. When additionally every term of the sum disappears

$$P_\alpha W(\alpha \rightarrow \beta) = P_\beta W(\beta \rightarrow \alpha) , \quad (1.7)$$

also detailed balance is fulfilled. This condition is usually employed to define the dynamics of an equilibrium system ad-hoc.

As a remark, note that in the systems studied in this thesis the concepts of energy and temperature do not apply intuitively. In the stochastic transport models discussed in Chapter 3 the properties of the system are determined by

the specific dynamics that define the system and the probability measure in the canonical ensemble is defined using the statistical weights of microstates which in turn depends on the dynamics. In the generation of complex graphs, discussed in Chapter 4, similarly statistical weights of configurations are used to construct the used statistical ensembles.

1.2. Non-Equilibrium Processes

Basically, there are two types of out-of-equilibrium systems: Those relaxing towards equilibrium (or the steady state) and those held out of equilibrium by an external field, forcing a current through the system. The systems of the former type are described by a Master equation with a non-zero left hand side and may be treated using approaches from linear response theory or by means of the fluctuation dissipation theorem if the perturbation is small enough. Systems of the latter type, exhibit a steady state, which is similar to an equilibrium state in several aspects. This becomes evident, as the Master equation of such systems becomes zero, i.e. the probabilities of the microstates are stationary.

A nonequilibrium system evolves through its local stochastic dynamics that do not fulfill detailed balance with respect to the state probabilities. Thus, the specific dynamics define the properties of the system. This is distinct to an equilibrium system, where the dynamics are usually defined in a way to give the proper statistical weights of states using the detailed balance condition.

It is a natural way to construct such a nonequilibrium system with a known steady state by introducing a current of a conserved quantity to an equilibrium system with specified dynamics. Thus detailed balance is broken and the dynamics are asymmetric. This can be accomplished by placing the system in between of two heat baths with different temperature, creating a heat current or by defining the dynamics as asymmetric motion of particles, creating a “mass” current.

The condensation process in the studied model is a combination of the mentioned type The specific properties of the studied non-equilibrium stochastic process are discussed in Chapter 3.

2. Numerical Methods

In this chapter I present the numerical methods used throughout this thesis. In particular these are methods used for the estimation of dynamic properties of the PFSS process and the computation of steady state and equilibrium quantity expectations of properties of dynamic systems or complex networks respectively. The former requires direct simulation of the stochastic process while the latter is rather accomplished by simulating the corresponding ensemble. In computational statistical physics these numerical methods have proven worthwhile since the availability of powerful computing machines and evolved in terms of precision as well as accuracy and thus effectivity.

The main problem in calculating expectation values of either steady state or dynamic properties is the vast size of the involved phase space. Every new site in the dynamic model will add one dimension to the phase space, thus increasing its size exponentially. Similarly the network configuration space grows exponentially with the size of the sampled networks.

The first two sections cover basic approaches to numerical computation and estimation of system properties, followed by a discussion of error estimation in this context. For a detailed introduction to Monte Carlo techniques in statistical physics, refer for example to [7]. In a fourth section, a generalization of the Hoshen–Kopelman cluster identification algorithm is discussed as the technical basis of several measurements in the estimation of the time scale of the condensation process.

2.1. Exact Enumeration

A first low-end approach to calculate the expectation of a quantity in a system, is to exploit the sheer computational power available today and integrate the configuration space. This can be done by realizing every single configuration \vec{m} of the system in order to calculate the expectation. It is useful to obtain the density of states Ω with respect to basic observables of the system while traversing the configuration space. For a Hamiltonian system, the Gibbs measure is used and the partition function reads

$$Z = \sum_{\{\vec{m}\}} \exp[-\beta E(\vec{m})] = \sum_E \Omega(E) \exp[-\beta E] \quad (2.1)$$

with the density of states involved. Just as well the canonic expectation calculates as

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \sum_E \langle \mathcal{O} \rangle_E \Omega(E) \exp[-\beta E] , \quad (2.2)$$

2. Numerical Methods

using the microcanonic expectation

$$\langle \mathcal{O} \rangle_E = \frac{1}{\Omega(E)} \sum_{\{\vec{m}\}} \mathcal{O}(\vec{m}) \delta_{E, E(\vec{m})}. \quad (2.3)$$

Of course, when treating non-Hamiltonian systems, the Boltzmann factor is replaced by an appropriate statistical weight with respect to the configuration and the used statistical ensemble. For the PFSS process this leads to expectation values of the form

$$\langle \mathcal{O} \rangle = \frac{1}{Z_N(M)} \sum_{\{\vec{m}\}} \prod_i \left[g(m_i, m_{i+1}) \delta_{\sum_i m_i, M} \right]. \quad (2.4)$$

Exact enumeration is indeed a feasible approach to compute quantities that are exact up to numeric errors for very small systems. Nevertheless, this method is quite useful to validate other methods, which has been done for the *graphgen* library discussed in Chapter 4. It is an important note, however trivial, that the exact enumeration of states or similar approaches do not allow the study of the dynamic properties of a stochastic process or out-of-equilibrium system. However, the estimation of such properties is required and will be described in section 2.2.2 below.

2.2. Monte Carlo Methods

As mentioned, exact enumeration is limited to small systems by the exponential growth of the configuration space. A solution to this is to merely strive for the computation of approximated property expectations. Introduced in 1949 by Metropolis and Ulam [8] the Monte Carlo method aimed at numerical integration of such exponentially growing spaces. It suggests that only a small subset of the original configuration space which is picked at random is used for the integration, while assuming that the chosen subset is representative. The benefit of using this statistical method is that, by the central limit theorem, the errors of estimates converge with $1/\sqrt{\mathcal{N}}$, where \mathcal{N} is the number of samples taken. This is opposed to other methods, where the rate of convergence highly depends on the dimension of the studied system and thus is far worse than $1/\sqrt{\mathcal{N}}$ for reasonably sized systems.

In the language of statistical physics this means to construct only a limited number of random configurations and use these to estimate any desired quantity of the system by means of statistical estimators. This method is known as simple sampling, as it provides no means to improve the representativity of the sampled subset of configurations. In fact, some highly probable configurations like ground states of the system might not be sampled at all in acceptable simulation time due to their small volume in state space. To compute accurate expectations nevertheless, there are two possibilities. Either the number of sampled configurations is increased to the order of magnitude of those in the original phase space, Or the method is modified to approximate or establish representative sampling of configurations. The former is of course out of question

and for the latter, there are many approaches designed to meet this requirement and improve convergence in general or specific situations. However, these are most often based on the Metropolis algorithm.

2.2.1. The Metropolis algorithm

The Metropolis algorithm [9] is such a method of selecting the system configurations to be sampled according to the probability distribution of the statistical ensemble. The strategy is to perform a random walk in the phase space where a step to the next configuration is performed by locally modifying the simulated system. For example, in the PFSS process this is done by moving a particle to another, arbitrary site in the system. However, it is important, that such a change of the configuration is not always carried out, but is accepted only with the probability

$$P_{\text{acc}}(\vec{m} \rightarrow \vec{m}') = \min \left\{ 1, \frac{P(\vec{m}')}{P(\vec{m})} \right\} \quad (2.5)$$

to establish the correct probability distribution. In this equation $P(\alpha)$ is the probability of a state α and the probability ratio for some Hamiltonian system reads $P(\vec{m}')/P(\vec{m}) = \exp[-(E(\vec{m}') - E(\vec{m}))/(k_{\text{B}}T)]$.

A local Metropolis update is carried out in two steps:

- (1) Propose a random modification to the configuration \vec{m} and determine the statistical weight of the new configuration \vec{m}' ¹²
- (2) Accept and make the proposed change $\vec{m} \rightarrow \vec{m}'$ with probability (2.5). The configuration is unchanged otherwise.³

By iteration of this update rule, a Markovian random walk in the configuration space is generated. From these states, observables \mathcal{O} can be measured and recorded as a time series for later evaluation. The expectations of most properties then calculate as simple averages of the corresponding time series of measurements.

It is important to note several remarks to properly apply this and most related sampling methods.

A sufficient condition for a Markovian random walk to correctly sample configurations with respect to their statistical weights is *detailed balance* (1.7) with the transition matrix elements $T_{\alpha\beta}$ given by (2.5) for transitions in the move set and $T_{\alpha\beta} = 0$ otherwise. However, detailed balance is fulfilled and trivial to check for the Metropolis algorithm (2.5).

¹The modifications proposed to graphs in the generation of networks graphs include the replacement of edges and are discussed in Chapter 4.

²When simulating Hamiltonian systems, it is sufficient to calculate the energy of the new configuration.

³This means that changes, that increase the statistical weight of the configuration (or decrease the energy) are accepted always, while changes lowering the statistical weight (increase the energy) are accepted with probability $W(\beta)/W(\alpha) < 1$ ($\exp[-\beta(E(\vec{m}') - E(\vec{m}))]$) for Hamiltonian systems).

2. Numerical Methods

To perform a correct simulation it is vital that the move set is *ergodic*. That is, the limited number of possible modifications considered at each step must allow the transformation of any configuration into any other by applying a finite chain of changes. Apparently this condition is fulfilled by the particle movement update used for the steady state simulation of the discussed particle systems, since any configuration can be obtained by at most M steps⁴. The case is more complicated for the graph updates discussed in Chapter 4, but still easy to see for most of the suggested updates. Nevertheless, ergodicity does not imply that every configuration is reached in reasonable time in a simulation. Consider for an example a system, where in order to reach a configuration a number of changes with very low acceptance probabilities must be taken. The simulation might then be trapped, incapable of passing such a region in configuration space. There are however simulation methods on top of Metropolis or otherwise that allow to pass such barriers. Examples are global move sets, such as cluster updates, generalized ensemble methods such as Multicanonic Monte Carlo or multiple Markov chain approaches such as parallel tempering. However, in the simulation of a process's dynamics, the forced passage of barriers is not desired.

Due to the nature of the Markovian random walk, the consecutively sampled configurations are highly *correlated*. This results in a reduced number of measurements that might be estimated by autocorrelation analysis and exploited for error estimation as discussed in Section 2.3 below. However it is sensible to reduce the actual frequency of measurements in the simulation by the order of \mathcal{N}_{eff} beforehand.

It is also reasonable to eliminate correlation of measurements to the initial configuration of the simulation by performing several MC steps until the sampling can be considered distributed as the steady state or equilibrium.

Again, for the explicit simulation of dynamics, the correlation of configurations is naturally desired and all measurements are recorded.

2.2.2. Dynamic Metropolis updates

Simulating the actual dynamics of a stochastic process of course means obeying the dynamic rules that define the process in the first place and a priori breaking detailed balance. So from the set of all possible Metropolis updates at any configuration, only a subset is allowed for proposal. In driven particle system like the PFSS process, only updates that move a particle to a neighbor site are valid. Then, also the acceptance rate for such an update is directly given by the defined dynamics. In driven particle systems like the ZRP this is apparently the properly normalized hopping rate function. It is evaluated at the proposed departure site with the current configuration to determine whether to move a particle to any of its neighbors.

The generated time series of system states represents a possible evolution of the dynamic system in time. This is opposed to the Markov chain of system states generated by standard Monte Carlo methods, that represents a random walk in the configuration phase. The correlation of measurements in the individual

⁴Perform M updates in steps: Take away all particles and put them at their new positions for the new configuration

time series is used to measure dynamic properties. Expectation values are then obtained by averaging over many distinct time series.

2.3. Error Analysis

There are several approaches to estimate the statistical errors of quantities estimated from experimental data or computer simulations. In this section I will give only a short introduction of some used methods. For a detailed overview see the introduction by Janke [10].

Due to the nature of the data generation a statistical analysis of errors is appropriate⁵. The standard error estimation method for experimental data employs the central limit theorem. The estimator $\bar{\mathcal{O}}$ of a quantity \mathcal{O} is

$$\bar{\mathcal{O}} = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \mathcal{O}_i, \quad (2.6)$$

with \mathcal{N} independent realizations \mathcal{O}_i . The error of a gaussian distributed $\bar{\mathcal{O}}$ is given in terms of the the standard deviation, or 1σ -confidence interval around the estimate, where σ is the square root of the variance. As the distribution of $\bar{\mathcal{O}}$ is related to that of \mathcal{O} via (2.6), the variance is $\sigma_{\bar{\mathcal{O}}}^2 = \sigma_{\mathcal{O}}^2/\mathcal{N}$. The latter can directly be estimated from the series of measurements to

$$\epsilon_{\bar{\mathcal{O}}} = \sqrt{\frac{\sigma_{\bar{\mathcal{O}}}^2}{\mathcal{N}}} = \sqrt{\frac{1}{\mathcal{N}(\mathcal{N}-1)} \sum_{i=1}^{\mathcal{N}} (\mathcal{O}_i - \bar{\mathcal{O}})^2}. \quad (2.7)$$

This method of error estimation is sufficient for most of the quantities that characterize the dynamic behavior of the PFSS process, as the simulation typically allows only one measurement per individual run of the condensation process.

Since simulations of the steady state or equilibrium ensembles generate Markov chains of configurations used to compute properties, consecutive measurements \mathcal{O}_i cannot be independent. This is fixed by estimating the typical time scale τ (in the time of the Markov process) at which the correlation of two measurements \mathcal{O}_i and $\mathcal{O}_{i+\tau}$ vanishes and they can be assumed independent. With this, the number of effectively independent realizations \mathcal{N}_{eff} of the quantity can be calculated and the error is estimated as

$$\epsilon_{\bar{\mathcal{O}}} = \sqrt{\frac{\sigma_{\mathcal{O}}^2}{\mathcal{N}_{\text{eff}}}}. \quad (2.8)$$

The time scale τ is obtained by computing the *autocorrelation function*

$$A(k) = \frac{\langle \mathcal{O}_i \mathcal{O}_{i+k} \rangle - \langle \mathcal{O} \rangle^2}{\langle \mathcal{O}^2 \rangle - \langle \mathcal{O} \rangle^2} \quad (2.9)$$

⁵Nevertheless, in some cases it is sensible to have constraints on numeric errors as well.

2. Numerical Methods

and evaluating the series

$$\tau = \frac{1}{2} + \sum_{k=1}^{\mathcal{N}} A(k) \left(1 - \frac{k}{\mathcal{N}}\right) \xrightarrow{\text{large } \mathcal{N}} \frac{1}{2} + \sum_{k=1}^{\mathcal{N}} A(k). \quad (2.10)$$

It is typically sufficient to evaluate the series only up to a self-consistent cutoff condition $k > 6\tau$. However, performing a full autocorrelation analysis to estimate errors is too elaborate, as more convenient methods are available for estimators. This is emphasized by the fact, that autocorrelation analysis does not allow easy application for any indirect measured quantity, that is a non-linear combination of direct measured properties. Nevertheless it is still a good method to estimate a useful frequency of measurements for a simulation.

The *binning* method is an approach to group the time series into (binning) blocks and apply standard estimation methods to a much shorter time series whose elements

$$\mathcal{O}_{B,n} = \frac{1}{k} \sum_{i=1}^k \mathcal{O}_{k(n-1)+i}, \quad k = \frac{\mathcal{N}}{\mathcal{N}_B} \quad (2.11)$$

are assumed to be independent. This relies on a sufficient block size k and again on the quantity not being a non-linear function of basic properties. The latter requirement is easily avoided by using the Jackknife method discussed below. Inserting this into (2.7), the binning error reads

$$\epsilon_{\bar{\mathcal{O}}} = \sqrt{\frac{1}{\mathcal{N}_B(\mathcal{N}_B + 1)} \sum_{j=1}^{\mathcal{N}_B} (\mathcal{O}_{B,j} - \bar{\mathcal{O}})^2}. \quad (2.12)$$

In the similar *Jackknife* analysis, instead of considering grouping the time series into blocks of size $k = \mathcal{N}/\mathcal{N}_B$ as above, \mathcal{N}_B blocks $\mathcal{O}_{J,n}$ are formed, each nearly as large as the original time series. These Jackknife blocks

$$\mathcal{O}_{J,n} = \frac{\mathcal{N}_B \bar{\mathcal{O}} - \mathcal{O}_{B,n}}{\mathcal{N}_B - 1} \quad (2.13)$$

contain the complete series of measurements, each with only a single binning block removed. The length of the individual blocks is now nearly that of the original time series. Thus, the bias of quantities, that are calculated as non-linear combinations of basic properties, becomes comparable to that obtained from the original data. With a correction of the trivial correlation due to widely overlapping data blocks the estimated error reads

$$\epsilon_{J,\mathcal{O}} = \sqrt{\frac{(\mathcal{N}_B - 1)}{\mathcal{N}_B} \sum_{j=1}^{\mathcal{N}_B} (\mathcal{O}_{J,j} - \bar{\mathcal{O}})^2}. \quad (2.14)$$

In the simulation of the PFSS process this form of error analysis is useful for the measurement of steady state properties. In the condensation, however, properties are time dependent anyway. As, in the simulation of the condensation process, quantities are averaged over many independent runs of the process, the individual measurements are uncorrelated and the error is estimated by variance of the respective quantity as in (2.7).

2.4. Cluster Identification

To numerically study properties of condensates and of their formation, a method to identify such structures of connected sites with at least $m_0 \geq 1$ particles each, is needed. It should apply to regular lattices and graphs as well. A prominent method used widely in the study of percolation models is known as the Hoshen–Kopelman (HK) algorithm [11]. It is usually defined and applied to regular lattices only, with several optimizations exploiting the fixed local structure. But it can easily be generalized to work on arbitrary graphs. Although graph theoretical approaches like spanning tree strategies fit the task quite naturally, they do not provide clear advantages over the Hoshen–Kopelman algorithm in either ease of application or performance scaling with system sizes⁶.

A cluster of sites is generally defined as a group of $n \geq 1$ sites belonging to the same equivalence class defined by a given criterion. The equivalence relation used to identify particle condensates is quite simple: A site is in the same cluster (condensate) as its neighbor, if both contain at least one particle $m_i, m_j \geq 1$.

A short description of the method employing a disjoint-set formalism is given below with Figure 2.1 explaining some of the specifics. Refer to [13] for a more detailed description. Basically, a disjoint-set data structure allows the representation of a partition of separate, nonoverlapping sets. It defines the union operations, to merge two sets into a single set, the find operation, to determine which set a given element is in and the trivial makeset operation, creating a set with a single element. This formalism allows the formulation of approaches to several partitioning problems.

To find every cluster in the system, a label L_i denotes the cluster to which a site i , or more general, another cluster with label i belongs. This leads to the identification of labels with sets of sites which are conditionally joined by the algorithm:

- (1) Initialize the list of labels. Every site is assumed to be a separate cluster and thus labeled individually $L_i = i$.
- (2) For every site, starting at the first site $i = i_0$, repeat steps
 - (2.1) Find the first neighbor n of i that is bonded to i and has already been visited ($n < i$)
 - (2.2) Find all other visited neighbors $n_j < i$ that have bonds with i and join the sets L_n and L_{n_j} by assigning the label $L_{n_j} = L_n$
 - (2.3) Join the sets of nodes n and i and assign the resulting set a cluster label $L_i = L_n$
- (3) Collapse all cluster labels. This is not necessary, if clusters are relabeled in depth during the find steps of the algorithm. That is, if a sequence of labels $L = (1, 1, 2, 3, \dots)$ is resolved to $L = (1, 1, 1, 1, \dots)$ immediately, which also speeds up further find steps.

⁶For a discussion of this method versus graph theoretical approaches in some detail see [12].

2. Numerical Methods

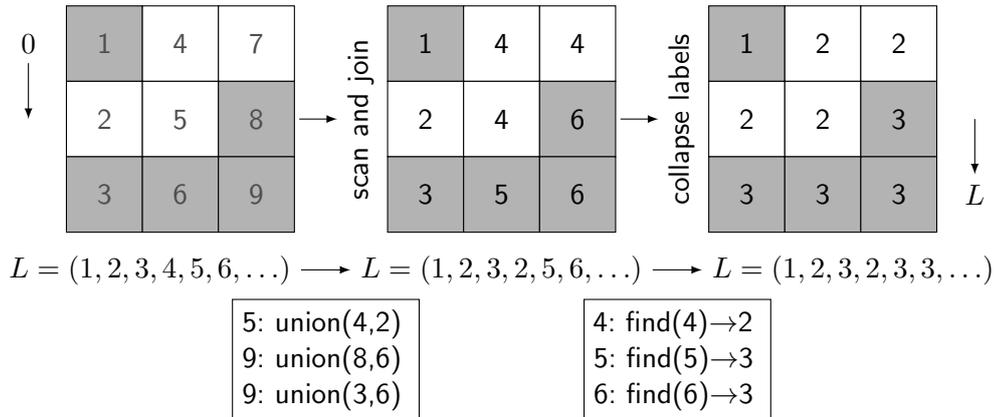


Figure 2.1.: Description of the Hoshen–Kopelman algorithm in the formulation of a disjoint-set data structure for a regular lattice. The procedure is started with initial labels. Already processed neighbor labels are unified if appropriate in a first step ((2.1) in the recipe). Then the common label is assigned to the site processed (step (2.2) in the recipe). Note that the union and find steps are typically run directly after another for every site. The step 5: union(4,2) means that being at site 5 the sets of neighbors 4 and 2 are joined to the set with label 2. For periodic boundaries, additional steps 3: find(3)→1 and 9: union(3,6) were executed.

After performing the single cluster identification run, while recording cluster masses in parallel, the sequence of labels encodes information on particle condensates in the system for this configuration.

3. Mass Condensation in Stochastic Transport

Stochastic mass transport describes the motion of a conserved generic mass. What is described in such a process by term “mass” may be manifold: passengers in public transport systems, macromolecules on the cytoskeleton [14], people browsing the world wide web or granular flow like sandpile dynamics [1]. In the following, the “mass” shall be represented by indistinguishable particles of unit mass. An interesting feature of these processes is that condensation of the “mass” may occur above a certain density, so that even in the limit of infinitely many particles a finite fraction of particles piles up as a bulk, although the dynamics may be fully symmetric.

The stochastic transport process studied in this thesis belongs to a class of models describing a gas of indistinguishable particles that are localized at sites of a lattice of network and can only move to their respective neighbor sites. This class of processes has been proposed by Spitzer [15] to study the properties of interacting Markov processes. It shall be noted, that these processes generally lack the usual concepts of temperature and energy, although these may be defined artificially by means of the statistical weights of the configurations. Nevertheless, many mappings to physical models have been found such as to the ones mentioned above, but even to models unrelated to stochastic transport, as quantum gravity and Bose–Einstein condensation [16].

These transport processes are in general held out of equilibrium by the current that drives the “mass” through the system. However, they may exhibit a steady state, that can be treated by methods similar to those used to study equilibrium systems. Such a transport process is constructed by taking a symmetric equilibrium transport process and forcing a current of the conserved “mass” through it. The result is a driven diffusive system.

I will begin this chapter with a description of a basic stochastic transport process that has no ranged interaction but nevertheless features the emergence of a condensate of particles. It is known as the zero-range process and serves as an introduction to the class of models and to basic estimation methods.

This will be followed by a description of the studied model which is similar to the above one but has a nearest-neighbor interaction. In the next section I will discuss the methods used to characterize the condensation process and estimate its time scale. In the fourth section, I present these results of the numerical studies.

3. Mass Condensation in Stochastic Transport

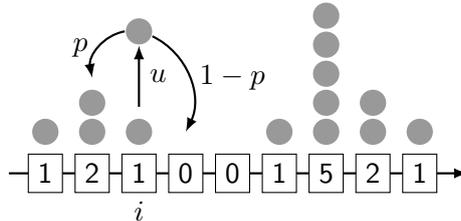


Figure 3.1.: Consider M particles initially distributed at random on N sites of a periodic chain. At a time step of the stochastic process, a random particle leaves a site i with probability u proportional to the hopping rate $u(m_i)$ to the left or right with probabilities p or $1 - p$ respectively.

3.1. A Zero-Range Transport Process

The zero range process (ZRP) is a simple stochastic mass transport process without ranged interaction.

The ZRP stochastic transport model assumes M particles in the system, initially placed at random on N sites. In the basic model the sites form a closed chain. Each site i contains a number $m_i \geq 0$ of particles up to M . This referred to as the occupation number of that site. The particles shall be indistinguishable and have unit mass. As of the basic model, the periodic boundary condition is $m_{N+1} \equiv m_1$. The specific dynamics are defined by the hopping rate $u(m_i)$. It gives the rate at which a particle leaves a site and jumps to one of the neighbor sites. The hopping rate depends only on the number of particles on the same site, hence the range of interaction is zero. After leaving a site, the particle might jump to the left with probability p and to the right with probability $1 - p$. This asymmetry corresponds to the effect of an external field driving the particles. An illustration of such a process is given in Figure 3.1 and guides the approach to simulation: A site i with with occupation $m_i > 0$ is randomly chosen and one particle is taken away with probability proportional to the hopping rate $u(m_i)$. The particle is then added to one of the neighbor sites. In the symmetric process this is randomly chosen, but the process can be tuned to the asymmetric case by introducing a probability p with which the particle goes to the left and otherwise to the right.

A most interesting feature of this stochastic process is the emergence of a condensate even though there is no ranged interaction between particles of different sites. However, when condensation occurs, the condensate cannot stretch and is localized at a single site due to the zero-range interaction. above a critical density ρ_c

The dynamics of the ZRP is governed by the hopping rate function and the asymmetry of the hopping. The variety of hopping rate functions and their effect on the system has been extensively studied in the past. When one choses the hopping rate, there are three main cases with respect to condensation in the

3.1. A Zero-Range Transport Process

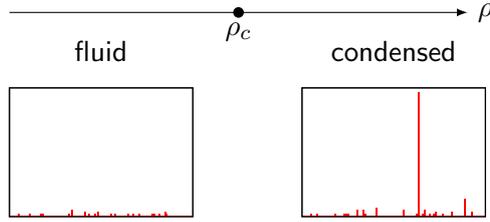


Figure 3.2.: Phases of the zero-range process. Below the critical density, the system is in a fluid state. Above the critical density a condensed phase of particles occupying a single site emerges.

limit of large systems [17]:

$$u(m_i \rightarrow \infty) = \begin{cases} 0 & \text{always condensed} \\ \text{finite} & \text{condensate may exist} \\ \infty & \text{fluid} \end{cases} . \quad (3.1)$$

Intuitively this means, that when $u(m \rightarrow \infty) = \infty$ there exists an effective repulsive force that prevents the occupation of a site by many particles. The particles will therefore be distributed uniformly on the chain. In the opposite case, where $u(m \rightarrow \infty) = 0$ there is an effective attractive force towards sites with many particles: The probability for a particle to leave such a site decreases the more particles occupy that site. This means that the critical density is $\rho_c = 0$. However, in the case where the hopping rate approaches a positive finite value u_∞ , one can show that the critical density becomes finite. This does not depend on the exact value of u_∞ , as it is a rate, but on how fast $u(m)$ approaches it. Therefore $u_\infty = 1$ can be fixed and a common choice for the hopping rate with tunable asymptotic behavior is used. In the rate

$$u(m) = 1 + \frac{b}{m}, \quad (3.2)$$

the asymptotic index b has this role. It is shown for example by Godrèchet [18], that for $b > 2$ the system has a finite critical density

$$\rho_c = \frac{1}{b-2} \quad (3.3)$$

where the correlation length diverges. If $0 \leq b \leq 2$, the attraction between particles is too low to form a condensate. A phase diagram with the typical configurations is displayed in Figure 3.2.

Above the critical density, the system is composed of a critical part, also referred to as the (critical) background and a condensate. The critical background consists of $\rho_c N$ particles, whereas the condensate contains the remaining $\Delta = M - \rho_c N$ particles. It is located at a single site due to the local interactions of particles.

3. Mass Condensation in Stochastic Transport

3.1.1. The steady state of the ZRP

To construct the steady state, symmetric dynamics are considered at first in order to obtain a system in equilibrium. Asymmetric hopping is introduced later without changing the steady state but with the consequence of a driven current.

Hence, to calculate the steady state probabilities, balance is assumed and the Master equation has the form

$$0 = \frac{1}{2} \sum_{i=1}^N \theta(m_i) u(m_i) P(m_1, \dots, m_i, \dots, m_N) \quad (3.4)$$

$$+ \frac{1}{2} \sum_{i=1}^N \theta(m_i) u(m_{i+1} + 1) P(m_1, \dots, m_i - 1, m_{i+1}, \dots, m_N),$$

where $P(m_1, \dots, m_i, \dots, m_N)$ denotes the steady state probability of the configuration $\vec{m} = (m_1, \dots, m_i, \dots, m_N)$ and the Heaviside function $heaviside(m_i)$ restricts the sum to configurations where there is a particle at site i to leave. This describes the conservation of probability currents due to the hops out of the left hand side configurations into the right hand side ones and vice versa is conserved. Now, with the factorized form

$$P(\vec{m}) = \frac{1}{Z(M, N)} \prod_{i=1}^N f(m_i) \quad (3.5)$$

of the steady state probabilities (with a weight function $f(m)$), the equality (3.4) for a single site i (with m_i presumably) simplifies to

$$u(m_i) f(m_i) f(m_{i+1}) = u(m_{i+1}) f(m_{i-1} - 1) f(m_{i+1} + 1) \quad (3.6)$$

by cancelling common weight factors. Rearranging the factors to the constant form

$$u(m_i) \frac{f(m_i)}{f(m_i - 1)} = u(m_{i+1} + 1) \frac{f(m_{i+1} + 1)}{f(m_{i+1})} \quad (3.7)$$

yields then the relation of the weight factor to the hopping rate function when setting the proportionality constant to 1. The resulting recursion $f(m_i) = f(m_i - 1)/u(m_i)$ gives then

$$f(m) = \prod_{n=1}^m \frac{1}{u(n)}, \quad f(0) = 1. \quad (3.8)$$

The resulting partition function of the steady state is then

$$Z(M, N) = \sum_{\vec{m}} \prod_{i=1}^N f(m_i). \quad (3.9)$$

3.1.2. Condensation process

In this section I turn to the dynamics of the condensation process and its time scale in this model. Starting with a random distribution of particles one observes that the dynamics can be divided into two regimes [19]. In a nucleation regime, the excess $M - \rho_c N$ particles accumulate at a finite number of sites until the density in the domains of sites in between reach the steady state density ρ_c . This is followed by a coarsening regime, in which the condensates exchange particles through the fluid background. This exchange leads to the evaporation of smaller condensates whose particles are absorbed by larger ones. This process continues until only a single condensate site remains while the average condensate mass *expectation* grows.

The time scale of the coarsening process, which dominates the condensation time is calculated by Evans [19] as

$$\tau \sim \begin{cases} L^3 & \text{in } d = 1 \\ L^2 \ln L & \text{in } d = 2 \\ L^2 & \text{in } d > 2 \end{cases} \quad (3.10)$$

for symmetric dynamics and

$$\tau \sim L^2 \quad (3.11)$$

in the asymmetric process. Besides a mean field approach, these time scales are predicted using a random walk argument discussed in [18, 20].

As a preparation and to test the estimation methods proposed in Section 3.3, I estimate this scaling behavior using numerical methods confirming the predicted time scale (3.11). An illustration of the evolution of the mass of the heaviest condensate and the number of condensates is given in Figure 3.3.

3.1.3. Condensation on inhomogeneous networks

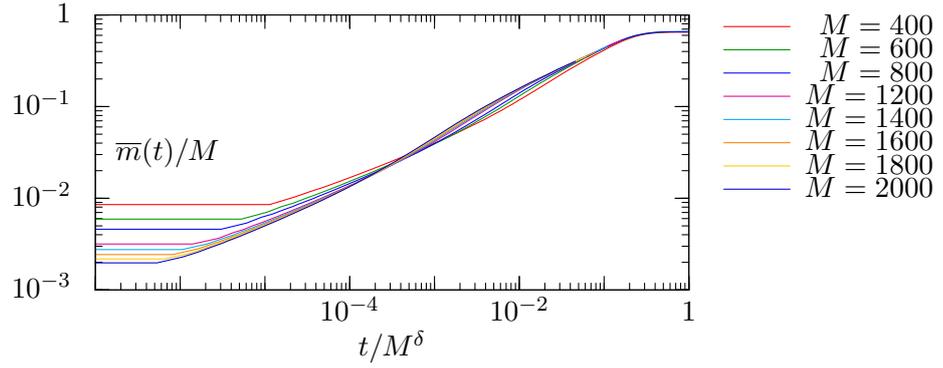
The zero-range-process can easily be adapted to networks. The difference is then, that the number of neighbors may be different at each site and is not limited anyhow. This means, for undirected networks, that no direction of movement is preferred, hence only the symmetric process is easily adopted. To study the asymmetric process, a preferred direction must be built into the — now directed — network. However, it has been shown that condensation in the symmetric ZRP also occurs on networks of different basic types [17, 21], such as star graphs and regular graphs with a single inhomogeneous site¹. Due to the inhomogeneous structure of these network types, this holds even for the trivial hopping rate

$$u(m) = \begin{cases} 0 & m = 0 \\ 1 & m \geq 1 \end{cases} \quad (3.12)$$

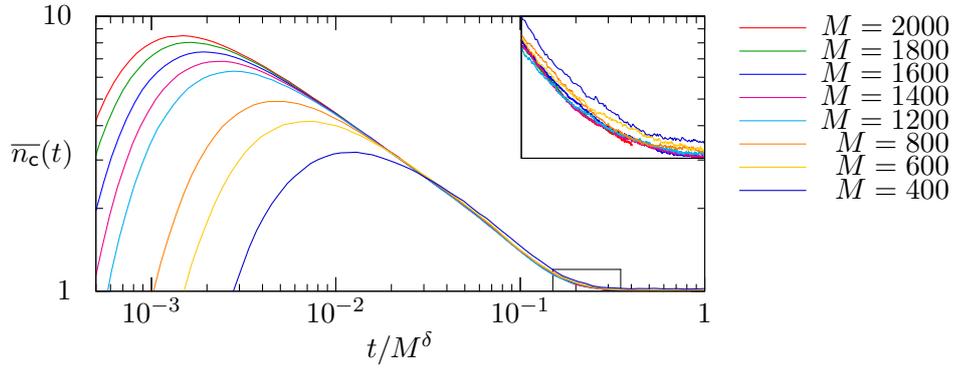
independent of the occupation at the departure site. This is shown, by calculating the distribution of occupation numbers in the system.

¹i.e. a regular graphs where one specific site is connected to all other sites.

3. Mass Condensation in Stochastic Transport



(a) $\delta \sim 2$



(b) $\delta \sim 2$

Figure 3.3.: Test of the scaling estimation method for the ZRP leads to the known exponent $\delta = 2 \pm 0.1$. In (a) the mass of the heaviest condensate and in (b) the average number of condensates ($m_i > 3/2\sqrt{M}$) are plotted versus rescaled time. The scaling exponent is found by matching the respective curves for different system sizes with $N = M$ sites averaged over $\mathcal{N} = 5000$ realizations.

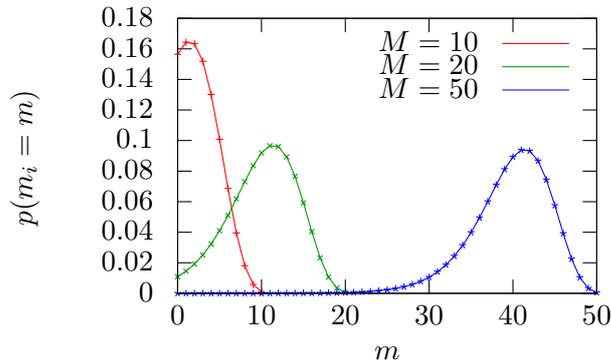


Figure 3.4.: Exact (solid lines) versus estimated (points) site occupation distribution in the ZRP on three different networks.

The steady state of the zero-range process on a graph reads [17]

$$Z(M, N) = \sum_{\{\vec{m}\}} \prod_{i=1}^N f(m_i) q_i^{m_i} \delta \left[\sum_i m_i - M \right], \quad (3.13)$$

with the site weight

$$f(m) = \prod_{n=1}^m \frac{1}{u(n)}, \quad f(0) = 1, \quad (3.14)$$

the degree sequence $\{q_i\}$ of the network and the Kronecker symbol $\delta[x] = \delta_{x,0}$. For calculations on the steady state of the process on scale-free networks the generic recursive form

$$Z(M, N) = \sum_{m_N}^M f(m_N) q_N^{m_N} Z(N-1, M-m_N) \quad (3.15)$$

can be simplified by exploiting the degree sequence. See Appendix 3.1.3 for further details. Using and evaluating the obtained recursion, the distribution of site occupation is computed for several small Barabasi–Albert networks with $N = 10, 20$ and 50 nodes respectively. The exact distribution is displayed in Figure 3.4 along with numerical data for comparison.

In [17, 21] it is also shown, that there is a typical time of the condensate to spontaneously disappear depending on the local structure and scaling with the condensate mass as a power law. This prediction, generalized to scale-free networks is confirmed numerically in [22].

3.2. Transport with Short Range Interactions and a Pair Factorized Steady State

The transport process proposed by Evans, Majumdar, and Hanney [23] is a generalization of the ZRP discussed above. Following them, I will also refer to this transport model as the pair-factorized steady state (PFSS) process. This naming is simply due to the fact that the steady state of the system factorizes over pairs of neighboring sites of the periodic chain. A benefit of this factorization is, that adaption to lattices or graphs is straight-forward, as bonds are fundamental building blocks of any structure.

The basic difference of this process to the ZRP is that the hopping rate function $u(m_i|m_{i+1}, m_{i-1})$ may depend on the number of particles at direct neighbor sites as well as the occupation of the departure site. Furthermore, it is assumed that the hopping rate factorizes to the form

$$u(m_i|m_{i+1}, m_{i-1}) = f(m_i, m_{i-1})f(m_i, m_{i+1}) \quad (3.16)$$

with a function $f(m, n) \geq 0$ for each nearest neighbor to get the PFSS and allow for a generalization of the process to networks.

This hopping rate introduces a short range interaction between the particles to the process. In particular, it is interesting to ask for the properties of the condensate and the condensation process when the interaction between particles tends to flatten out the site occupation profile and therefore that of the condensate. In the following the steady state of the system is constructed and it is shown that the weight factor proposed in [23] gives such a hopping rate.

3.2.1. The steady state of the PFSS process

In the following it is shown, that a system with factorized hopping rate (3.16) has the pair-factorized steady state

$$P(\vec{m}) = \frac{1}{Z(M, N)} \prod_{i=1}^N g(m_i, m_{i+1}) \delta_{M, \sum_{i=1}^N m_i} \quad (3.17)$$

with the normalization $Z(M, N)$ that is similar to the partition function in equilibrium systems and a symmetric weight function $g(m, n)$ that fulfills

$$f(m, n) = \frac{g(m-1, n)}{g(m, n)}. \quad (3.18)$$

The Kronecker symbol $\delta_{M, \sum_{i=1}^N m_i}$ in (3.17) ensures the correct number of particles in a configuration. In the following, its shall be abbreviated as δ_M .

The procedure to show this is similar to that used for the ZRP. The evolution of the state probabilities $P(\vec{m}, t)$ in time are described by the Master equation

$$\frac{dP(\vec{m}, t)}{dt} = \left[\sum_{\vec{m}'} W(\vec{m}' \rightarrow \vec{m}) P(\vec{m}', t) - W(\vec{m} \rightarrow \vec{m}') P(\vec{m}, t) \right]. \quad (3.19)$$

3.2. Transport with Short Range Interaction and PFSS

The system assumes the steady state, when the state probabilities become stationary $P(\vec{m}) = P(\vec{m}, t \rightarrow \infty)$, i.e. the left hand side vanishes and the balance condition

$$\sum_{\vec{m}'} W(\vec{m}' \rightarrow \vec{m}) P(\vec{m}') = \sum_{\vec{m}'} W(\vec{m} \rightarrow \vec{m}') P(\vec{m}) \quad (3.20)$$

remains. The transition rates $W(\vec{m} \rightarrow \vec{m}')$ are non-zero only if there is a transition $\vec{m} \rightarrow \vec{m}'$ with a single hop, that is either $\vec{m}' = (\dots, m_{i-1} + 1, m_i - 1)$ or $\vec{m}' = (\dots, m_i - 1, m_{i+1} + 1, \dots)$. Then, the transition rate is given by the hopping rate $u(m_i | m_{i-1}, m_{i+1})$. For a fixed state \vec{m} there are therefore two possible states \vec{m}' per site that can be reached and the sums in (3.20) can be rewritten using the site index as

$$\begin{aligned} & \sum_{i=1}^N P(\vec{m}) u(m_i | m_{i-1}, m_{i+1}) = \\ & \sum_{i=1}^N \left[p P(\dots, m_i - 1, m_{i+1} + 1, \dots) u(m_{i+1} + 1 | m_i - 1, m_{i+2}) \right. \\ & \quad \left. + (1 - p) P(\dots, \dots) u(m_{i-1} + 1 | m_{i-2}, m_i - 1) \right]. \end{aligned} \quad (3.21)$$

The two parts in the right hand side refer to the cases where the configuration \vec{m} is reached by moving a particle to the right with probability $1 - p$ and to the left with p respectively. Now, using the assumption (3.16) and the guessed connection to the weights (3.18), the hopping rate reads

$$u(m_i | m_{i-1}, m_{i+1}) = \frac{g(m_i - 1, m_{i-1}) g(m_i - 1, m_{i+1})}{g(m_i, m_{i-1}) g(m_i, m_{i+1})}. \quad (3.22)$$

The proposed steady state probabilities (3.17) and the obtained hopping rate can then be inserted into the steady equation (3.21) leading to

$$\begin{aligned} & \sum_{i=1}^N \prod_{j=1}^N g(m_j, m_{j+1}) \delta_M \frac{g(m_i - 1, m_{i-1}) g(m_i - 1, m_{i+1})}{g(m_i, m_{i-1}) g(m_i, m_{i+1})} = \\ & \sum_{i=1}^N \left[p P(\vec{m}) \frac{g(m_{i-1}, m_i - 1) g(m_i - 1, m_{i+1} + 1) g(m_{i+1} + 1, m_{i+2})}{g(m_{i-1}, m_i) g(m_i, m_{i+1}) g(m_{i+1}, m_{i+2})} \right. \\ & \quad \times \frac{g(m_{i+1}, m_i - 1) g(m_{i+1}, m_{i+2})}{g(m_{i+1} + 1, m_i - 1) g(m_{i+1}, m_{i+2})} \\ & \quad + (1 - p) P(\vec{m}) \frac{g(m_{i-2}, m_{i-1} + 1) g(m_{i-1}, m_i - 1) g(m_i - 1, m_{i+1})}{g(m_{i-2}, m_{i-1}) g(m_{i-1}, m_i) g(m_i, m_{i+1})} \\ & \quad \left. \times \frac{g(m_{i-1}, m_i - 1) g(m_{i-1}, m_{i-2})}{g(m_{i-1}, m_i - 1) g(m_{i-1} + 1, m_{i-2})} \right]. \end{aligned} \quad (3.23)$$

Using the symmetry $g(m, n) = g(n, m)$ to cancel several weight factors and introducing the abbreviation $R_i = \prod_{j \neq \{i-1, i, i+1\}} g(m_j, m_{j+1})$ for the common

3. Mass Condensation in Stochastic Transport

weight factors, the equality reads

$$\begin{aligned} \sum_{i=1}^N &= g(m_i - 1, m_{i-1})g(m_i - 1, m_{i+1})g(m_{i-1}, m_i)R_i \\ &\sum_{i=1}^N \left[p g(m_{i-1}, m_i - 1)g(m_i - 1, m_{i+1})g(m_{i+1}, m_{i+2})R_i \right. \\ &\quad \left. + (1 - p) g(m_{i-1}, m_i - 1)g(m_i - 1, m_{i+1})g(m_{i+1}, m_{i+2})R_i \right]. \end{aligned} \quad (3.24)$$

Both sides are now equal and it is proved that the hopping rate of the form (3.22) leads to the pair-factorized steady state (3.17). As the steady state factorizes over bonds of the structure on that the process takes place, the proof can easily be enhanced to lattices or an arbitrary graph [24]. In this case, the hopping rate (3.22) assumes the factorized form

$$u(m_i | m_{\langle i,k \rangle}, m_{\langle i,l \rangle}, \dots) = \prod_{\langle i,j \rangle} \frac{g(m_i + 1, m_j)}{g(m_i, m_j)}, \quad (3.25)$$

where the notation $\langle i, j \rangle$ denotes that the site j is a direct neighbor of i .

It is convenient to define the hopping rate function in terms of the local weight function $g(m, n)$ using (3.22). A factorization of the weight

$$g(m, n) = K(|m - n|) \sqrt{w(m)w(n)} \quad (3.26)$$

into a local interaction factor $K(|m - n|)$ between particles at neighboring sites and a zero range interaction $w(m)$ is proposed in [25] and indeed useful for the study of different interactions. The function $w(m)$ is in fact the same as the weight factor $f(m)$ in steady state of the ZRP and by setting $K(|m - n|) = 1$ constant it is also the ZRP that is retrieved.

However, in this thesis I stick to the choice of weights proposed by Evans [23]

$$g(m, n) = \exp \left[-J|m - n| + \frac{1}{2}U(\delta_{m,0} + \delta_{n,0}) \right], \quad (3.27)$$

that give the functions

$$K(x) = e^{-Jx}, \quad w(m) = e^{U\delta_{m,0}} \quad (3.28)$$

in the factorized representation (3.26). From the steady state probabilities

$$P(\vec{m}) = \exp \left[-J \sum_{i=1}^N |m_i - m_{i+1}| + U \sum_{i=1}^N \delta_{m_i,0} \right] \quad (3.29)$$

one can now see, that differences in the particle occupation at neighboring sites are exponentially suppressed. This means, that the dynamics prescribed by the weights (3.28) yield the desired feature, namely the tendency to flatten out the site occupation profile. Therefore, the parameter J can be interpreted as a kind of surface stiffness. The second term $U \sum_{i=1}^N \delta_{m_i,0}$ gives a repulsive on-site

3.2. Transport with Short Range Interaction and PFSS

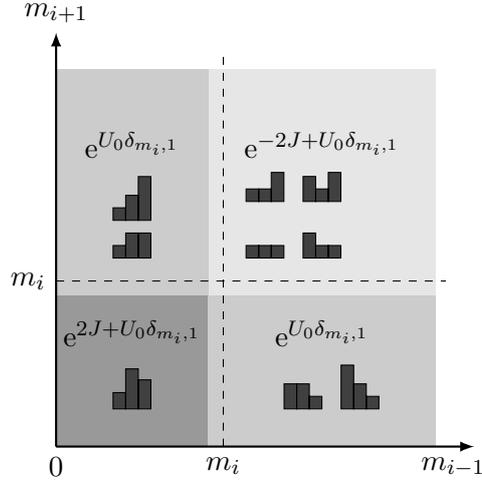


Figure 3.5.: Values of the hopping rate $u(m_i|m_{i-1}, m_{i+1})$ using the weights (3.28) for different occupation patterns. Darker grey refers to higher hopping rates. The axes m_{i-1} and m_{i+1} refer to the number of particles at the left and right neighbor site respectively. Due to the exponential form of $K(x)$ and the ratio (3.22), there are only the cases (3.30) split for $m_i = 1$ and $m_i > 1$ respectively. In higher dimensions or graphs, the values have similar cases.

potential. The larger the parameter U is, the more suppressed are sites occupied by a single particle. As in the ZRP, it is indeed the zero-range interaction, that leads to the separation of a liquid and a condensed phase. The effect of this term can therefore be interpreted as that of a hydrophobic substrate to droplets of water.

With the choice of the weights (3.29) the hopping rate function reduces to the cases

$$u(m_i|m_{i-1}, m_{i+1}) = \begin{cases} e^{2J+U_0\delta_{m_i,1}} & m_i > m_{i\pm 1} \\ e^{-2J+U_0\delta_{m_i,1}} & m_i \leq m_{i\pm 1} \\ e^{U_0\delta_{m_i,1}} & \text{otherwise} \end{cases} \quad (3.30)$$

due to the absolute difference of the particle numbers at neighbor sites. A display of these cases with the possible configurations is shown in Figure 3.5. This simplicity of the dynamics allows for a better understanding and greatly reduces the computational effort needed to simulate the process, as the rates do not have to be calculated each time.

3.2.2. Properties in the steady state

In order to study the properties of the steady state, the same methods of statistical physics as for equilibrium systems can be used. The steady state probability $P(\vec{m})$ can be interpreted as the probability of a microstate and the

3. Mass Condensation in Stochastic Transport

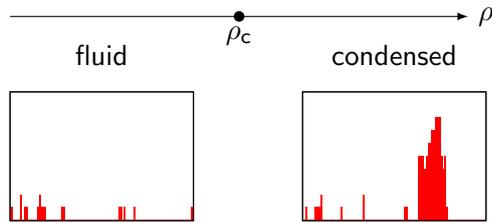


Figure 3.6.: Phases of the PFSS process with typical configurations. Below the critical density the system is in a fluid state with single particles or small droplets driven through the system. At the critical density a condensed phase emerges while the fluid phase stays critical.

canonical partition function

$$Z(M, N) = \sum_{\vec{m}} \prod_{i=1}^N g(m_i, m_{i+1}) \delta_M \quad (3.31)$$

can be defined just as well as the grand-canonical one

$$Z_N(z) = \sum_M Z(M, N) z^M = \sum_{\vec{m}} z^{\sum_{i=1}^N m_i} \prod_{i=1}^N g(m_i, m_{i+1}), \quad (3.32)$$

where z is the fugacity determined by

$$\rho = \frac{1}{N} \left\langle \sum_{i=1}^N m_i \right\rangle = \frac{z}{N} \frac{\partial \ln Z_N(z)}{\partial z}. \quad (3.33)$$

The grand-canonical partition function $Z_N(z)$ is monotonically increasing with z as well as the density (3.33). It is shown in [23, 26], that if $Z_N(z)$ has an infinite radius of convergence the site occupation distribution

$$p(m) = \frac{1}{N} \left\langle \sum_{i=1}^N \delta_{m, m_i} \right\rangle \quad (3.34)$$

decreases with m and therefore the system is always in a homogeneous fluid state. Otherwise, if $Z_N(z)$ has a finite radius of convergence z_c , the density can either grow to infinity or go to a constant in the limit $z \rightarrow z_c$. In the latter case it is shown that a critical density exists above which the grand-canonical ensemble does not exist, indicating a phase transition from the fluid to the condensed state. This means, that above the critical density ρ_c the system separates into a critical fluid phase with density ρ_c and a condensate containing the excess particles $\Delta = M - \rho_c N$. The Figure 3.6 gives an overview of the phases with typical states of the system.

In the following, I will give an overview of several properties of the system in its steady state. For a detailed discussion and derivation of these and other properties, also for other choices of weights, refer to [23, 26].

3.2. Transport with Short Range Interaction and PFSS

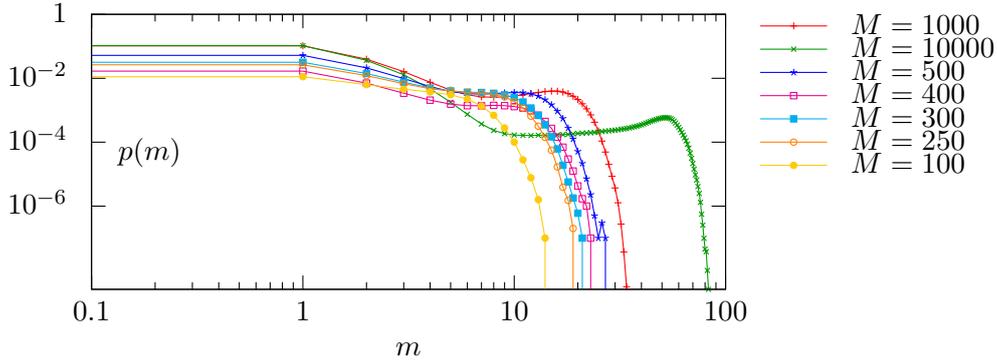


Figure 3.7.: Distribution of the site occupation in the steady state of the PFSS process for several system sizes. The systems up to the size of $M = N = 500$ were used to estimate the time scale of the condensation process. The particle distributions for larger systems ($M = N = 1000$, $M = N = 10000$) are added to allow the comparison of finite size effects to the condensation.

The critical density of the one-dimensional system with the used weights is calculated as $\rho_c \approx 0.2397$. The site occupation distribution (3.34) gives the probability that a randomly chosen site is occupied by m particles. It can be evaluated in the steady state using a transfer matrix method, either by numerical diagonalization or for some choices of weights analytically. This quantity can be easily measured by recording a histogram of site occupations. The site occupation distributions of the systems, that were used to study the condensation time scale are displayed in Figure 3.7 besides the distributions of some larger systems. From this data it is already visible, that the systems of sizes that are feasible to simulate still have large finite-size effects that affect the steady state and likely the condensation process.

The envelope shape of the extended condensate is calculated under the reasonable assumption, that the zero-range factor in the weights is neglectable. To accomplish this the expectation of the shape is rewritten to new variables $d_k = m_k - m_{k-1}$ and evaluated by means of a generating function using a saddle point approximation. The envelope shape reads

$$h(x) = \frac{w}{2v} \ln \left(\frac{\cosh J - \cosh vt}{\cosh J - \cosh v} \right), \quad (3.35)$$

where the rescaled variables

$$h(x) = \frac{\langle m_n \rangle}{\sqrt{M'}}, \quad x = \frac{2n}{w\sqrt{M'}} \quad (3.36)$$

are used. For the weights used in this thesis, the constants are calculated as $v = 0.5413$ and $w = 2.2005$ [25]. From the rescaled variables one can see, that the width of the condensate scales as the square root of the number of excess particles in the system. The average condensate for finite systems and the exact shape are displayed in the Figure 3.8.

3. Mass Condensation in Stochastic Transport

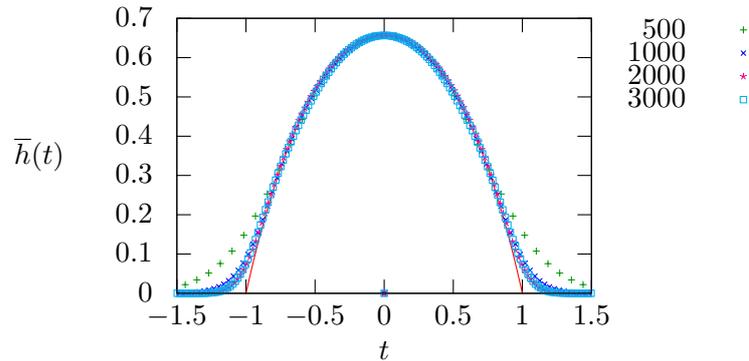


Figure 3.8.: Average envelope shape of condensates in systems with $N = 1000$ sites and $M = 500, 1000, 2000, 3000$ particles respectively. The solid line represents the exact shape in the limit of infinite systems (3.35).

However, it is shown in [25, 26] that the shape of the condensate is non-universal and depends on the form of the interaction term $K(x)$ of the statistical weight. For example, when considering the exponential form $K(x) \sim e^{-|x|^\beta}$ the envelope can be tuned to rectangular shape by lowering the parameter β . Additionally the width of the condensate might be adjusted from a single site to the extension of $\sqrt{M'}$ via the zero range part of the weight.

3.2.3. Dynamics of the condensation

In this section, I will discuss a description of the condensation and propose several mechanisms that could explain the process and its time scale.

As for the zero-range process, the condensation is observed to occur in several stages:

1. Initially many small droplets emerge from the homogeneous distribution of particles, as the background relaxes to its critical fluid state. This is the first regime in the condensation process.
2. The small droplets evaporate relatively fast and other condensates absorb the excess particles until only two or three large condensates remain. This is the beginning of the coarsening regime.
3. The system with the remaining condensates coarsens further until a single large condensate that contains all excess particles is left.

These stages can be observed in the provided Figure 3.9 which shows exemplary time series recorded in with $N = 1000$ sites and $M = 3N = 3000$ particles. It is also observable, that the time scale of each of the stages is much larger than that of the previous stage. Therefore the condensation time scale of the system is clearly dominated by the coarsening process.

In the study of this process the following condensation mechanisms were considered to take place in the coarsening regime:

3.2. Transport with Short Range Interaction and PFSS

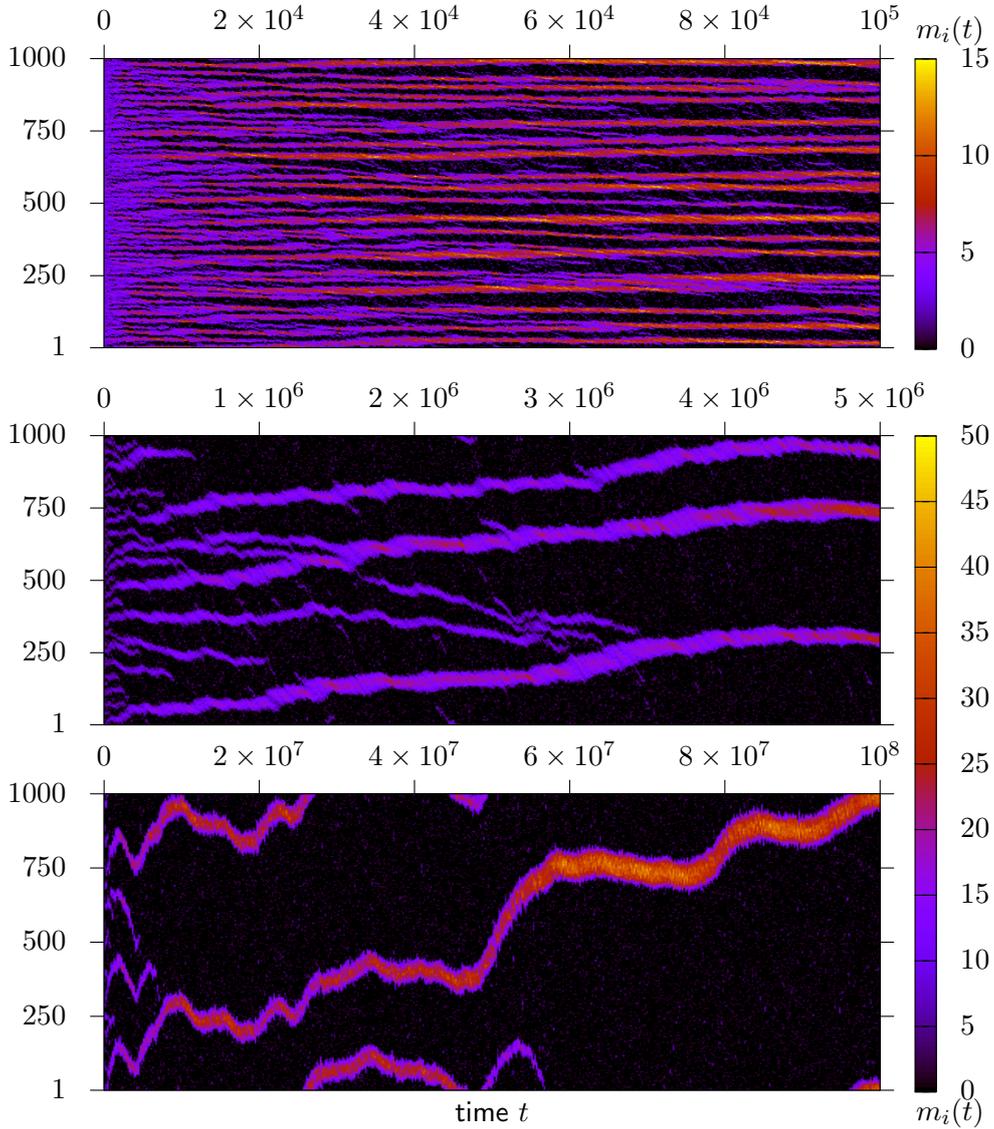


Figure 3.9.: Typical evolution of the condensation process on different time scales for the totally asymmetric ($p = 1$, particles hop in negative i direction) model. The displayed time series were recorded in a system with $N = 1000$ sites and $M = 3000$ particles prepared with a homogeneous distribution of particles. In the plot at the top, the emergence of a finite number of small droplets is observed in the early stages of the condensation process, which is referred to as the first regime. These droplets have a range of widths of 10 to about 50 sites and mass of 20 to 200 particles at the time of 10^5 MC sweeps. In the middle plot, this situation is seen at the left before the coarsening process begins. In a first stage droplets grow to larger condensates due to the fast evaporation of smaller droplets. However, the coarsening regime is dominated by the last stage, where only two condensates remain. It is also quite notable, that an effective long range interaction between large condensates affecting their movement is observed.

3. Mass Condensation in Stochastic Transport

- Movement of the condensates.

The condensates perform a possibly biased random walk in the system and eventually collide. In this case, the condensation time scale would be given by the rate of movement of the condensates, possibly depending on their size and the first passage time of a (biased) random walk. This should be directly observable in the time series of the evolving systems. The rate of movement can then be estimated by the movement of the center of mass of the condensate.

- Mass-dependent evaporation condensates.

If the current of particles leaving a condensate decreases with growing mass M' of the condensate, then for small condensates the current of particles escaping it is smaller than the current into the condensate. The dependence of the estimated current on condensate mass can then be used to calculate the condensation time scale.

- Fluctuation of the condensate masses.

The mass of the condensates in the coarsening regime is considered to perform a random walk. The time scale is that of first passage time as in the random walk argument used in prediction of the condensation time scale of the ZRP. In this case, the expected time scale is that of the ZRP.

However, the first proposed mechanism of condensation through collision of condensates can be abandoned right away, as in many observed time series like Figure 3.9 such events are very rare.

In the next section, after a brief discussion of methods used to estimate the properties in the steady state I will describe the methods employed to numerically estimate the time scale independent of the mechanisms and study properties of the currents out of and between the condensates to find evidence for the condensation mechanism.

3.3. Estimation methods

3.3.1. Properties of the steady state

The steady state properties are estimated by simulating the steady state with a standard Metropolis method instead of following the system dynamics.

To calculate properties of the condensate, such as its mass and shape, it is necessary to perform a cluster identification before. In this thesis the method described in Section 2.4 has been used to accomplish this task, but that method is in general replaceable. With the system separated into a heavy condensate and a number of small droplets and single particles it is easy to calculate the condensate mass M' and size v_c of a condensate just by counting the particles and sites in the according cluster respectively. It shall be noted, that these quantities are in principle sensible to the used definition of the condensate

borders that is used. In this thesis, any particles in a cluster are considered part of the condensate up to the sites at the borders that contain no particles².

The critical density of particles in the system can be estimated indirectly using the average mass of the condensate its size v_c . It is defined by the number of excess particles

$$\Delta = M - \rho_c N \quad (3.37)$$

that are contained in the condensate. However, the average mass M' is not identical to the number of excess particles as they build up above the background to form the condensate. The number of excess particles is therefore estimated by subtracting the average number of particles in the background below the condensate from the measured mass $M' = \Delta + \rho_c v_c$ using the number of sites v_c occupied by the condensate. Inserting this into (3.37), the critical density becomes

$$\rho_c = \frac{M - M'}{N - v_c} . \quad (3.38)$$

The envelope shape of the condensate is estimated by finding the heaviest condensate in the system and recording its shape to a histogram. Due to the movement of the condensates, the measured shapes must be translated according to their respective centers of mass before recording. The obtained average shapes are displayed in the Figure 3.8. However, this method does not work to estimate the average condensate shape on graphs since the translation would be undefined. Also, the condensate shapes would be specific to the particular graph structures, thus making it difficult to sensibly take an average over different graph realizations. The related distribution of site occupation numbers is measured simply by recording a histogram of m_i and has been displayed in Figure 3.7.

The current in the whole system is measured by keeping a histogram of successful updates and tried updates per site. The obtained average current measured with this method is $\bar{j} \sim 0.0544 \pm 9 \times 10$. However, it is clear, that this method allows the measurement of the current of particles on a per-site basis. To do such a per-site measurement the properties of the system, like the position of the condensate, have to be held stable, as the measurement would be blurred otherwise. This is accomplished by preparing the system very near to the state, where the measurement is taken and allowing the system to relax for a short time without measurement. Then, again only for a short time, the system can be simulated while measuring the per-site currents. This procedure must be repeated until enough measurements have been taken. The meaning of a “short time” depends of course on the state of the system in which the measurement is taken: If the system shall be in its final condensed state, the measurement may be relatively long, but if the system is prepared with two condensates as shown in Figure 3.10 only several hundreds MC sweeps (full system updates) are acceptable before the condensate masses change noticeably.

²Another acceptable definition would be to set the borders of a condensate to the sites, where the occupation drops below or equal to the (critical) density of the background. However, with the used weights, this is identical to the used distinction.

3. Mass Condensation in Stochastic Transport

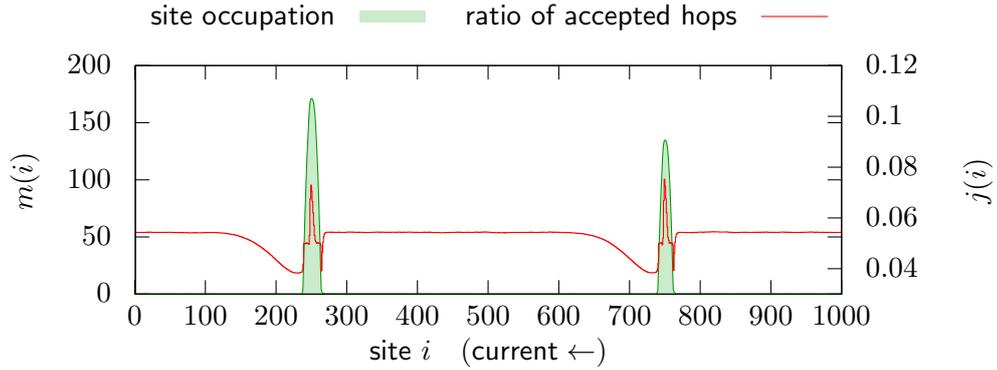


Figure 3.10.: The average particle current per site in the steady state is estimated in a configuration that is prepared with two condensates and critical background. In the system of size $N = 1000$ sites two condensates containing $M'_1 = 2000$ and $M'_2 = 3000$ particles respectively with added critical background are prepared. The total amount of particles is $M = 5000 + \rho_c N = 5239$ particles. To avoid changes of the properties of the condensates the data is averaged over $\mathcal{N} = 1000$ independent simulations of very short duration (several hundreds MC sweeps for bringing the prepared system to the steady state and for measurement respectively). The current has several interesting points characterizing how particles move into, over and out of the condensate. The sharp dip and maximum in the current when the particle reaches the condensate and crosses its top respectively are understood with a glance at Figure 3.5. The same goes for the dip at the condensate border when the particle leaves it. However, the most interesting feature of the current is at the sites directly after leaving the condensate. The current performs a steady relaxation back to the average current in the fluid phase. This is indeed a feature, that is not observed in the symmetric process and therefore characteristic to the process being out of equilibrium.

3.3.2. Estimation of the condensation time scale

I will first describe the numerical methods used to generally estimate the time scale of the condensation time scale. The basic difference between these methods is which indicator of the final condensed state is detected. As only a single time measurement can be taken per complete simulation and per estimator, the amount of data is expected to be small for acceptable system sizes. Therefore the usage of different different estimation methods is imminent in order to check the consistency of the results and allow for a better judgement of the statistical errors. It is practical to carry out all of the following methods but the last in the same batch of simulations, as computational costly tasks, like cluster identification are needed for each of them.

The simplest method is to use a mass threshold m_{thresh} as the indicator. When the mass of the largest condensate passes this threshold as $M' > m_{\text{thresh}}$, the time is recorded. A good choice for the threshold is the number of excess particles $\Delta = \rho_c N$, as the condensate mass is likely to pass it just before reaching its expected mass and thus the steady state. The simulation is repeated and at the end the average condensation time is calculated.

The second method also uses the mass of the heaviest condensate as an indicator. For the largest condensate, the average mass $M'(t)$ at time t is calculated in many simulations of the condensation process. As this recorded time series of masses is a representation of the condensation progress, it evolves in the same time scale. When this time series is known for different system sizes, one can plot them versus rescaled time to determine the scaling of the condensation time.

With an appropriate definition, the number of condensates in the system can be measured. When the average number of condensates n_c is measured at any time t , the result is a similar indicator to the condensation progress as the time series $M'(t)$ discussed above. As a distinction, whether a bulk of particles should be counted to this number of condensates or not, a mass threshold is proposed. The choice of this threshold is a tradeoff that affects the signal-to-noise ratio in the time series. If it is too low, fluctuations in the fluid phase are often counted, if it is too high the recorded time series is flattened. The mass of the droplets at the beginning of the coarsening regime, which is in the order of \sqrt{M} is a good starting point to define that threshold. Additionally, the average number of condensates is recorded with respect to several thresholds $a\sqrt{M}$ resulting in one time series per prefactor a .

Furthermore, in a similar approach as the first method, the first-passage time to reach a single condensate $n_c = 1$ according to the above definition can be recorded in a histogram. The average condensation time can then be calculated directly from the data or obtained by fitting an adequate probability density function.

Now I turn to the methods used to investigate, which of the proposed mechanism of the coarsening process can be ruled out or confirmed. These are the measurement of the influence of the condensate mass on the effective current of particles escaping the condensate and a method to compare the currents between two condensates of different masses.

3. Mass Condensation in Stochastic Transport

The first of these methods involves keeping account of the condensate masses present in the system and can be done simultaneously to the methods for the general estimation of the time scale. Before each MC sweep, a table containing the masses and positions of the present condensates and their borders is updated. Then at every successful update at one of the border sites, i.e. when a particle hops out of a condensate, this is recorded for the corresponding condensate mass. This implies the assumption, that the condensate does not move or change its shape considerably during one MC sweep. After many simulations of the process, the result is an estimate of the effective current of escaping particles for every condensate mass. As the coarsening process takes the longest time in the condensation, it is expected, that the most precise estimates of this current will be for masses of the order of the total number of particles. To reduce the computational cost of this measurement and avoid noise, only the two largest condensates are used in this method. Furthermore, it shall be noted that the current that is estimated with this method does not compare numerically to directly estimated currents due to the approach of the measurement. The motion of the condensate is not corrected and the measurement site is not guaranteed to be at the border of the condensate at all times. Nevertheless, the methods will prove to represent the influence of the condensate mass to the rate of its evaporation well enough.

To compare the currents between condensates of different sizes, the method to estimate per-site currents from the last section is used. Therefore this estimation method stands separated from the above. The system is prepared with two condensates of the desired mass and a critical fluid background and shortly simulated for “equilibration” before a short measurement of the currents at two sites in between. This is repeated many times to obtain precise estimates. It is useful to observe the occupation profile of the system to be able to repeat the measurement if the shape and mass of the condensates change considerable. In this case the measurement time has to be decreases further.

3.3.3. Remarks for the simulation on lattices and graphs

Most of the methods discussed above can just be used to estimate the properties of the PFSS process on two-dimensional lattices or arbitrary graphs as well. However, the measurement of the effective current of escaping particles becomes much more complicated because the borders now consist of many sites. Additionally, the assumption of slowly changing borders is less likely to hold. Currents between condensates may be estimated by extending the measurement of per-site currents to measure the amount of particles passing “membranes” between condensates.

Also, the consistent formulation of asymmetric dynamics is a nontrivial task at least for networks. A reasonable approach is to use directed graphs. When a particle hops, it chooses a leaving edge of its departure node and follows it to occupy the target node.

However, as the time to study the condensation process on networks was limited, I restricted myself to a brief investigation of the symmetric process on two-dimensional lattices and $k = 4$ regular graphs. These graphs were

generated using the graphgen-toolkit, that was created for that purpose and will be discussed in Chapter 4. Specifically this means, that starting with a square lattice the edges were rewired to generate random regular graphs. The lattices used for simulation as well as the initial square lattice graphs have periodic boundaries.

3.4. Results

3.4.1. Condensation on the periodic chain

In the following I will discuss the data and estimates of the condensation time obtained by the methods described in the last section. Most of this data was obtained for the asymmetric PFSS process, for several different system sizes and a constant particle density of $\rho = 1$. However, the estimates have been compared with data generated in simulations of systems with density $\rho = 3$. Table 3.1 gives an overview of the used system sizes and the number of condensation processes simulated for that system size.

Table 3.1.: Simulation effort for different system parameters of the one dimensional systems. The columns are the number of sites N , length of individual runs t_{\max} in MC sweeps and number of individual runs \mathcal{N} .

N	t_{\max}	\mathcal{N}	N	t_{\max}	\mathcal{N}
50	1×10^5	5×10^5	50	1×10^6	1×10^4
75	1×10^5	6×10^5	75	1×10^6	1×10^4
100	2×10^5	5×10^5	100	1×10^6	1×10^4
125	2×10^5	1×10^5	125	1×10^6	1×10^4
150	2×10^5	1×10^5	150	4×10^6	1×10^4
200	1×10^6	5×10^4	175	4×10^6	1×10^4
225	1×10^6	5×10^4	300	5×10^6	2×10^3
250	1.5×10^6	1×10^5	350	5×10^6	4×10^3
275	1.5×10^6	1×10^4	400	5×10^6	2×10^3
300	2×10^6	2×10^4	(b) $\rho = 3$		
325	2×10^6	1×10^4			
350	2×10^6	2×10^4			
375	3×10^6	1×10^4			
400	3×10^6	2×10^4			
500	5×10^6	1×10^4			

(a) $\rho = 1$

The average first passage times of the largest condensate mass passing a threshold are displayed in 3.11 for several system sizes along with the same property measured for the ZRP. The observed condensation times roughly follow the power law

$$\tau \propto M^\delta \tag{3.39}$$

3. Mass Condensation in Stochastic Transport

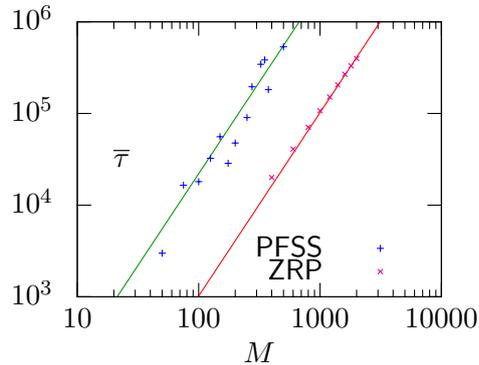


Figure 3.11.: Average first passage time of the heaviest condensate's mass exceeding the mass threshold $M'_{\text{thresh}} = \Delta = M - \rho_c N$ for the ZRP and PFSS process respectively. The solid lines represent the scaling obtained by other methods.

with the scaling exponent $\delta \sim 2$, which gives the same scaling as for the zero-range process. In the following, this will be confirmed by the other estimation methods which also prove to give much more precise estimates for the scaling exponent δ .

Both methods of recording the time series of the largest condensate mass and the number of condensates are analyzed in a very similar way. The time series for different system sizes are drawn in one plot versus rescaled time according to the presumed scaling law (3.39). The scaling exponent δ is then estimated by finding a value for which the curves of different system sizes collapse. To properly collapse the curves and estimate the correct scaling exponent, the emphasis lies in the behavior near to the point where the steady state is reached. That is, while approaching the final condensate mass and a condensate count of one respectively. In the case of the time series of condensate masses, also the values $M'(t)$ have to be rescaled to $M'(t)/M'(t \rightarrow \infty)$ using the mass of the condensate $M'(t \rightarrow \infty)$ in the steady state. The resulting plot resulting from this procedure is displayed in Figure 3.12. This rescaling is apparently not needed for the time series of the average number of condensates, as the systems approach the steady state with $n_c(t \rightarrow \infty) = 1$, if the mass threshold used for condensate counting is just high enough. However, in order to get usable results, the mass threshold has to be chosen lower, resulting in $n_c(t)$ approaching a finite value $n_c(t \rightarrow \infty) > 1$ as bulks of particles always emerge and melt in the background due to fluctuations. Otherwise the resulting time series become very flat, which makes collapsing the curves hard and unprecise. The resulting rescaled time series of the average number of condensates for several system sizes and different mass thresholds $a\sqrt{M'}$ are shown in the Figures 3.13 and 3.14.

The scaling exponents estimated using this method of collapsing the rescaled time series according to the presumed scaling law confirm the rough estimation from above and consistently provide boundaries for it. The estimated scaling exponents are the following:

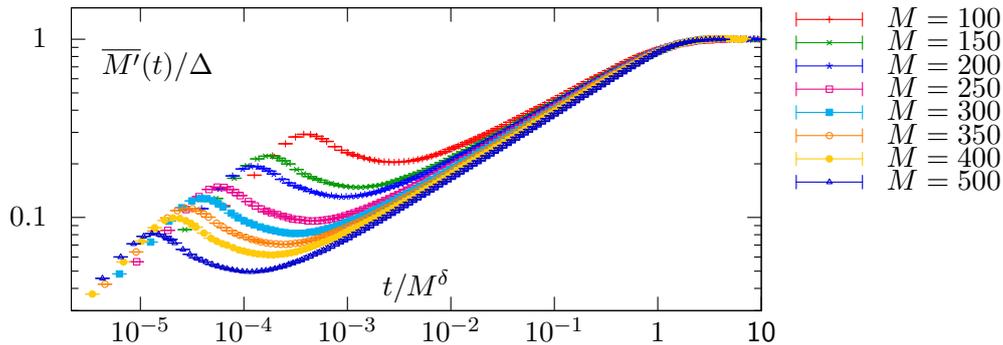


Figure 3.12.: Time scale analysis of the condensation process with respect to the average mass of the heaviest condensate at the time t . The curves of the condensate mass for different system sizes $N = M$ are matched by rescaling time according to the presumed scaling law (3.39). The mass is rescaled to the final mass of the condensate $M'(t \rightarrow \infty)$ and time is rescaled to t/M^δ , giving an estimation of the scaling exponent $\delta = 2.1 \pm 0.2$.

Largest condensate mass	$\delta = 2.1 \pm 0.2$
Number of condensates, $\rho = 1$	$\delta = 2.0 \pm 0.2$
Number of condensates, $\rho = 3$	$\delta = 2.0 \pm 0.1$

Using the rescaled time series of the condensate mass, also the scaling exponent of the symmetric process is estimated as $\delta = 2.9 \pm 0.2$. Again, this is similar to the scaling exponent $\delta = 3$ of the symmetric ZRP.

Additionally to estimates of the scaling exponent, these rescaling methods also give a good impression of the evolution of the measured quantities in the coarsening process. In both time series, there seems to be condensed state at the beginning of the simulations. This is in fact an artefact caused by the homogeneous distribution of particles at the beginning: Very broad, but flat (in terms of occupation numbers) clusters are identified before the coarsening regime of the condensation process. However, when the coarsening regime begins, the growth of the mass of the largest condensate approaches the curve $1 - \exp(-t/\tau)$.

The validity of these rescaling methods has been tested for the ZRP giving the correct scaling exponents. The rescaled time series of the condensate mass and number of condensates obtained in these tests are displayed in the Figures 3.3a and 3.3b respectively.

By the next method, the number of condensates measured in every simulation sweep was used to measure the first passage time, when this number becomes $n_c = 1$ in that individual condensation process. The formation of small condensates in the background can be neglected in this approach, as the time scale on which this happens is very small in comparison to the coarsening time scale. The obtained distributions of first passage times for several system sizes are shown in Figure 3.15. The shape of the normalized histograms suggests that the

3. Mass Condensation in Stochastic Transport

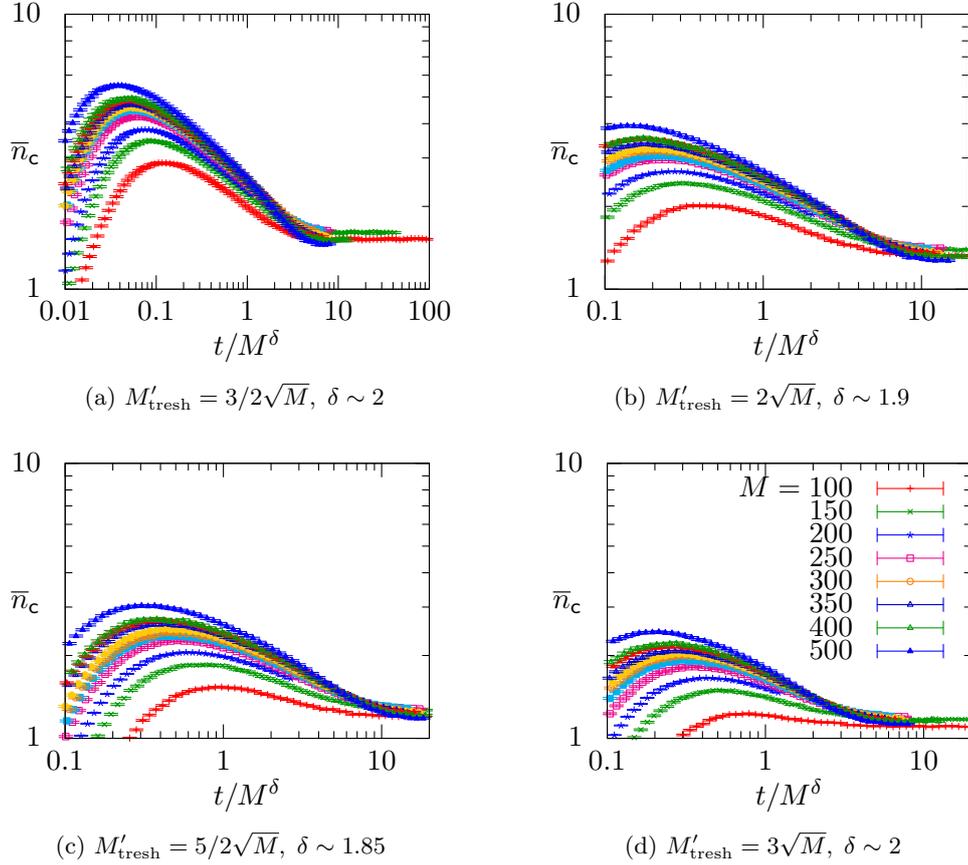


Figure 3.13.: Time scale analysis of the PFSS process using the number of condensates \bar{n}_c with mass M' greater than a fixed threshold mass M'_{thresh} . The time is rescaled to t/M^δ to match curves and determine the scaling exponent $\delta \approx 2$. From top to bottom, the mass threshold is increased from $3/2\sqrt{M}$ (a) to $3\sqrt{M}$ (d). The system sizes are $N = 100, 150, 200, 225, 250, 275, 300, 325, 350, 375, 400$ and $N = 500$, from bottom to top at the left of each graph respectively. The number of particles is $M = N$ for all curves. To estimate $\delta = 2.0 \pm 0.2$, the curves are rescaled until they match in the region where they fall into the horizontal. However, the uncertainty of the determination of δ by this method is considerable, especially since the curves cannot be collapsed exactly due to different final values of $\bar{n}_c(t \rightarrow \infty)$.

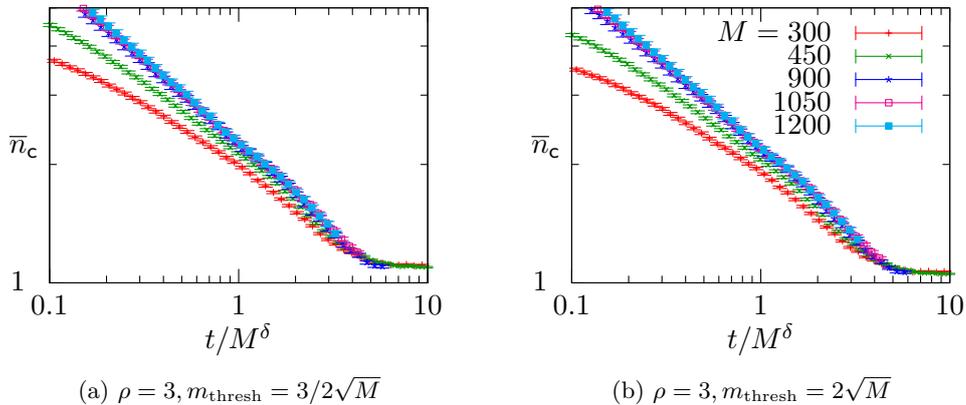


Figure 3.14.: Time scale analysis of the PFSS process using the number of condensates \bar{n}_c with mass m_c greater than a fixed threshold mass m_{thresh} as in Figure 3.13. However, these plots represent data from systems with a higher density $\rho = 3$. The same scaling exponent $\delta = 2 \pm 0.1$ is obtained. Note, that the data presented in this plots is accumulated in only several thousand condensation runs due to the fast growth of the condensation time and therefore simulation time.

first passage times are distributed according to an extreme value distribution. Indeed, the data fits well with a Gumbel-type extreme value distribution

$$f(x) = \frac{ze^{-z}}{\beta}, \quad z = \exp\left[-\frac{x - \mu}{\beta}\right] \quad (3.40)$$

where μ is the central peak position, and β is the width parameter. The average condensation time can now be easily estimated by fitting the distribution and calculating its mean value $\tau = \mu + \beta\gamma$, where $\gamma \approx 0.57721566$ is the Euler–Mascheroni constant. The obtained mean condensation times are now plotted in Figure 3.16, again confirming the presumed scaling law (3.39) and giving a more precise estimation of the scaling exponent

$$\text{First passage time to single condensate} \quad \delta = 2.04 \pm 0.07.$$

To summarize, the presumed scaling law (3.39) is confirmed by different approaches to estimate the condensation time scale and the scaling exponent is estimated to be $\delta = 2.04 \pm 0.7$, $\delta \approx 2$ within good approximation. This is the same scaling of the condensation time as that of the ZRP and the similarity of the scaling could also be confirmed for the symmetric PFSS process and ZRP exhibiting a scaling exponent of $\delta = 3$. It should be noted, that for the system sizes that were feasible for computer simulation still strong finite size effects exist. However, from the obtained data it can be seen, that in the range of studied system sizes of $N = M = 50$ to $N = M = 500$ the rescaled curves of measured properties tend to approach their generic form. For example, from Figure 3.7 and comparison to the shape of the exact distribution [26] one can

3. Mass Condensation in Stochastic Transport

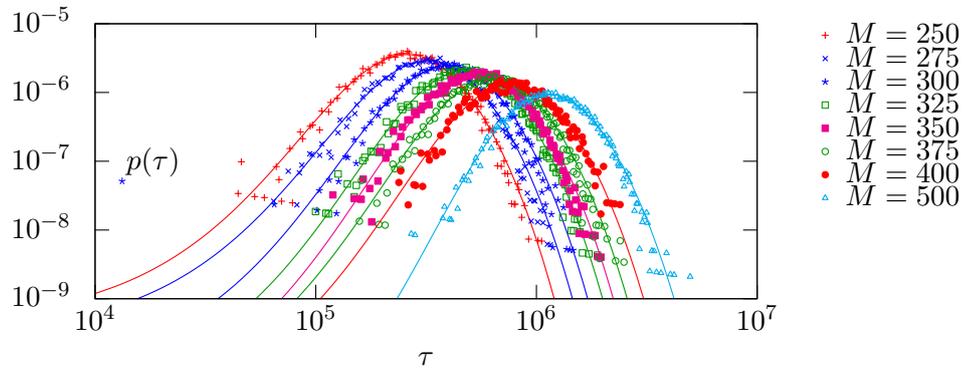


Figure 3.15.: Distribution of first passage times for the condition $n_c = 1$, for a mass threshold of $5/2\sqrt{M}$. That is, when the number of condensates first drops to 1. The solid curves represent the fitted Gumbel-type extreme value distribution (3.40).

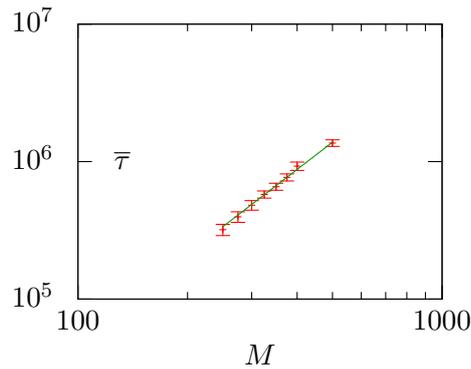


Figure 3.16.: Using the distributions of first passage times in Figure 3.15, the scaling exponent of the first passage time for $n_c = 1$ is estimated as $\delta = 2.04 \pm 0.07$.

see, that the distribution does not assume this shape until a system size of $N = 1000$.

In the following I will discuss the results obtained by the measurement of the currents in the system. In the assumed mechanism of the longest lasting stage of condensation, only two large condensates are left in the system. As there is no merging of condensates in real space observed, I will start with the discussion of the second proposed mechanism: mass dependent evaporation of the condensates.

The effective current of particles escaping a condensate of mass M' has been measured for several system sizes by the discussed method and is plotted in Figure 3.17 versus the rescaled condensate mass M'/M . It can be observed, that the current decreases relatively fast with the mass for small condensates $M' < 0.3M$ and then decreases more slowly until a shallow minimum is reached for the expected mass of the condensate in the steady state. For high masses of the condensate, the number of events is very small resulting in large errors of the estimated current. Although the measured current cannot necessarily be numerically compared to the real current of particles escaping the condensate, the existence of the minimum in the measured current shows that it represents the features of that current and its dependence on the condensate mass well enough.

Now to the question whether the time scale of the coarsening process can be explained by the measured currents. In the last regime of the coarsening process which dominates the time scale, two condensates remain in the system. The assumed evaporation leads to two currents $j_e(M'_1)$ and $j_e(M'_2)$ of particles between the condensates with mass M'_1 and M'_2 . In the asymmetric process the particles cross the fluid background between the condensates in a time that is of the order of the number of sites N and is therefore small in comparison to the condensation time scale. As also the escaping particles current depends only weakly on the condensate masses, this time can be neglected and the current between the condensates is conserved. That is, the rate equation

$$\frac{dM'_1}{dt}(t) = -j_e(M'_1(t)) + j_e(\Delta - M'_1(t)) \quad (3.41)$$

should describe the growth of the condensate M'_1 . To determine the typical time scale of this process, one can rescale the time to the condensation time scale $t' = \Delta^\delta t$ and the condensate mass to units of total excess particles giving $M'_1 = \Delta x$ and $M'_2 = \Delta(1 - x)$. The rescaled rate equation reads then

$$\frac{1}{\Delta^{\delta-1}} \frac{dx}{dt'}(t') = -j_e(\Delta x) + j_e(\Delta(1 - x)) . \quad (3.42)$$

A solution to this rate equation must make it dimensionless and therefore have a factor $1/\Delta^{\delta-1}$ itself. Note, that although here $\tau \propto \Delta^\delta$ is used as the time scale, this is different to the presumed scaling law (3.39) just by a constant

3. Mass Condensation in Stochastic Transport

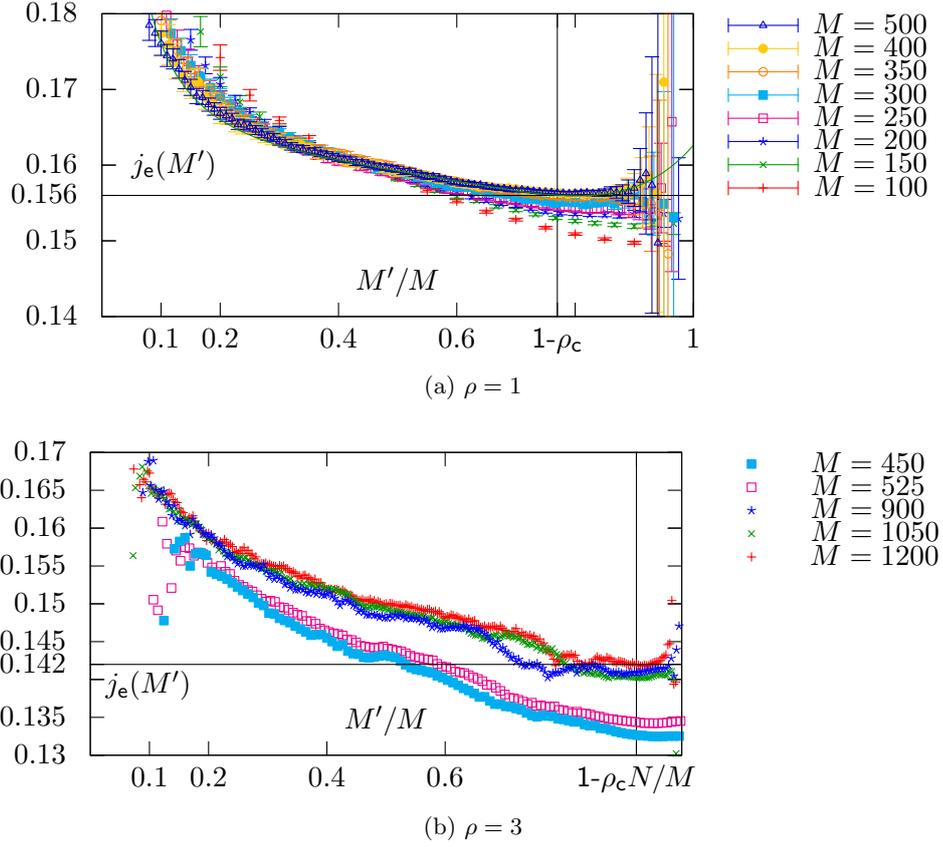


Figure 3.17.: Escape flow rate j_e plotted versus the size M' of the condensate a particle escapes. The particle density in (a) is $M = N$ and $M = 3N$ in (b), while the latter is provided as a supplement (as the data base is rather poor). However, the minimum of the escape flow at $M'/M = 1 - \rho_c L/M$ supports the fact of the existence of a single condensate with a steady mass of $M - \rho_c N$. The solid line in (a) is the approximated current (3.43).

factor $(1 - \rho_c)^\delta$. The equation (3.42) means in order to reproduce the observed time scale with the estimated scaling exponent $\delta = 2$, the current has the form

$$j_e(M') = j_0 + \frac{b}{\Delta} f\left(\frac{M'}{\Delta}\right), \quad (3.43)$$

with a constant current j_0 , an arbitrary function $f(x)$ of the fraction of excess mass in the condensate to approximate the form of the estimated current and the factor $1/\Delta$ making (3.42) dimensionless. b is a parameter to fit the proposed current to the measured one.

The function $f(x)$ is determined by guessing a polynomial to fit the rescaled current. The choice

$$f(x) = 1 + c\left(x^8 - \frac{3}{4}x^4\right) \quad (3.44)$$

has been found to fit very well with similar parameter sets for different system sizes. These parameters are

$$j_0 \approx 0.154, \quad b \approx 0.8, \quad c \approx 9.$$

This fit is also shown in Figure 3.17 with the estimated currents for comparison. However, to see if the form (3.43) can be used in principle, one can rescale the estimated currents to the shape function

$$f\left(\frac{M'}{\Delta}\right) = \frac{\Delta}{b} (j_e(M') - j_0). \quad (3.45)$$

This should collapse the estimated curves to the shape $f(x)$ independent of the system size. Figure 3.18 shows that this is not fulfilled. This means, that it is not likely that this approach can explain the coarsening process and its time scale alone. However, it must be stressed that the estimated current $j_e(M')$ is too noisy to make a definitive decision. This may be answered by further research using better data. Nevertheless, since already the time scale of the condensation has been observed to be the same as in the ZRP, it is a likely approach to translate the random walk argument to the PFSS process.

In this approach, the fluctuations of the condensate mass are considered to be stronger than the currents j_e . In the coarsening process, the masses of the condensates then perform a biased random walk in the interval $0, \dots, M$. This bias to the fluctuation of the mass is given by the current j_e . The first passage time of the random walk to reach $M' = \Delta$ then gives the condensation time scale. This in turn depends on the strength of the bias, which is studied by estimating the difference of the currents between two condensates of different mass as discussed in the last section. The obtained differences of these currents for condensate pairs of different masses are shown in Figure 3.19. They are not only too small to cause a significant bias and thus the coarsening in the prepared systems but are also scattered around zero within acceptable range of error. This means that the bias is indeed negligible and the time scale is that of a random walk, namely $\delta = 2$.

3. Mass Condensation in Stochastic Transport

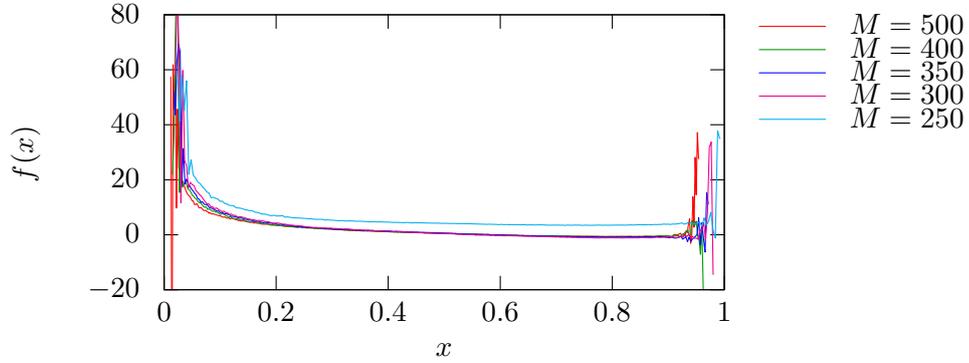


Figure 3.18.: The rescaled shape of the current of particles escaping a condensate of a given mass (according to (3.44)) depends on the parameters of the system. Therefore this approach can only partially explain the coarsening process and its time scale.

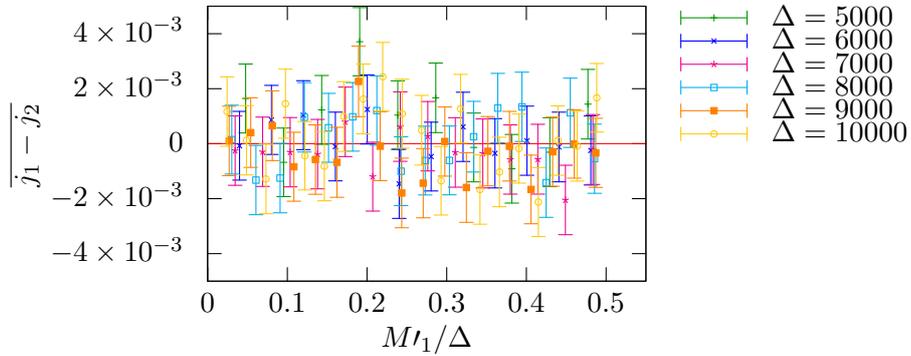


Figure 3.19.: Estimation of the difference of background currents $j_1 - j_2$ between two condensates of moderate masses M_1' and $M_2' = \Delta - M_1'$ respectively. Apparently, as the differences of the currents are scattered around zero within the error of the estimation, the hypothesis of condensation through mass dependent currents does not hold for comparable masses of the remaining condensates.

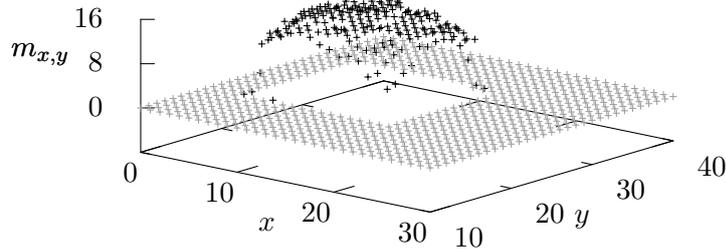


Figure 3.20.: Shape of a condensate on a two dimensional lattice of size 50×50 sites with $M = N = 2500$ balls.

3.4.2. Condensation on Regular and Disordered Lattices

The condensation process and time scale in the one-dimensional process has been shown to be very similar to the ZRP. If this also holds for the two-dimensional case, a scaling exponent of $\delta = 2$ is expected for the symmetric and asymmetric process as well.

The critical density in the PFSS process on the square lattice is much lower than in one dimension. It is estimated as $\rho_c = 0.00317 \pm 0.00005$ in a simulation of the steady state. The distribution of particles $p(m = m_i)$ is very similar to that observed for the system in one dimension. The profile of the shape of the condensate, shown in Figure 3.20, is comparable to that of the one-dimensional process and can be approximated as a simple power law of fourth order.

The condensation time scale has been estimated by rescaling the time series of the largest condensate mass and the number of condensates with the resulting plots shown in Figures 3.21 and 3.21b respectively. The scaling exponent is estimated as

Largest condensate mass	$\delta = 2.0 \pm 0.1$
Number of condensates	$\delta = 1.9 \pm 0.1$

confirming the exponent predicted for the ZRP. The available data was insufficient to estimate the scaling exponent with greater precision using the first passage times to reach a single condensate state.

The study of the PFSS process on graphs has been limited to a brief survey for regular graphs. These were generated using the graphgen toolkit and for each graph the condensation process has been simulated once in order to average over different disordered regular graphs. Before the estimation of the condensation time scale, the critical density was measured to learn whether condensation occurs and if so what the quantity of excess particles in the following simulations will be. The critical density was estimated employing the steady state simulation of the process on $\mathcal{N} = 1000$ regular graphs (with $k = 4$ nearest neighbors per site) with the result $\rho_c = 0.14 \pm 0.02$.

When estimating the condensation time scale of the PFSS process by rescaling the time series as shown in Figure 3.22 one finds, that there seems to be no scaling of the condensation time with the size of the system at all. The rescaled

3. Mass Condensation in Stochastic Transport

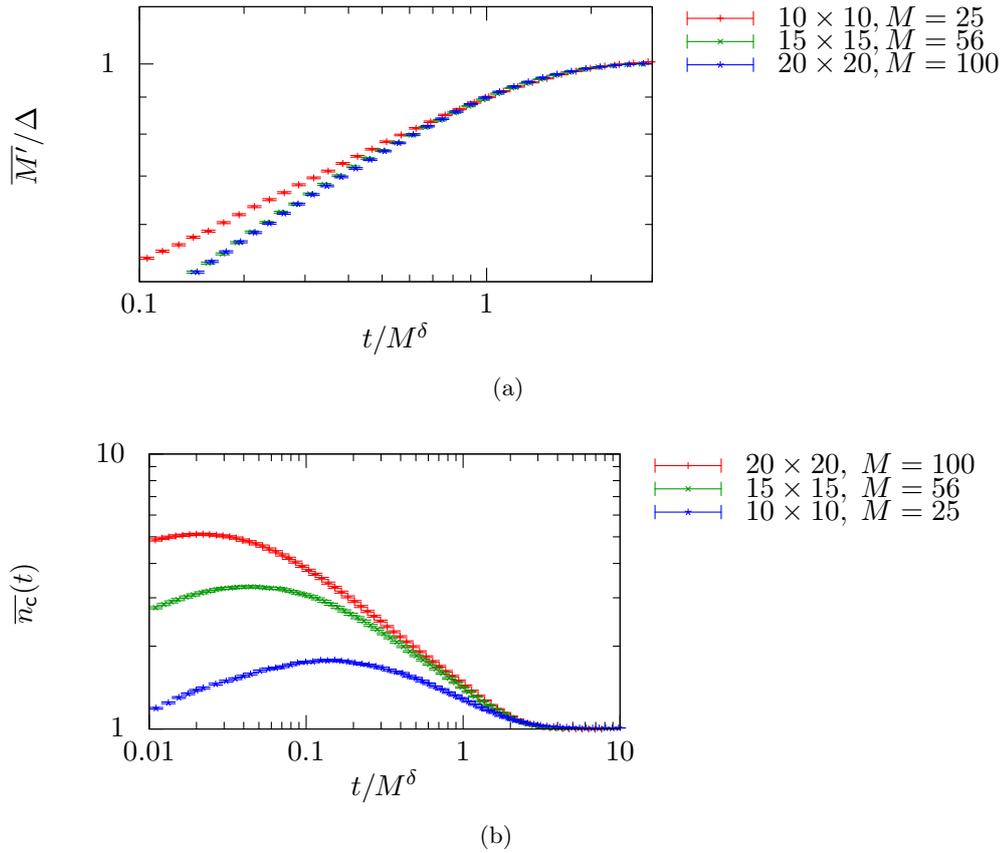


Figure 3.21.: Estimation of the condensation time scale for the symmetric PFSS process on a two-dimensional lattice with periodic boundaries. (a) The collapsing rescaled curves of the heaviest condensate mass yield a scaling exponent of $\delta = 2 \pm 0.1$. (b) Rescaling the number of condensates $n_c(t)$ gives a scaling exponent of $\delta = 1.9 \pm 0.1$.

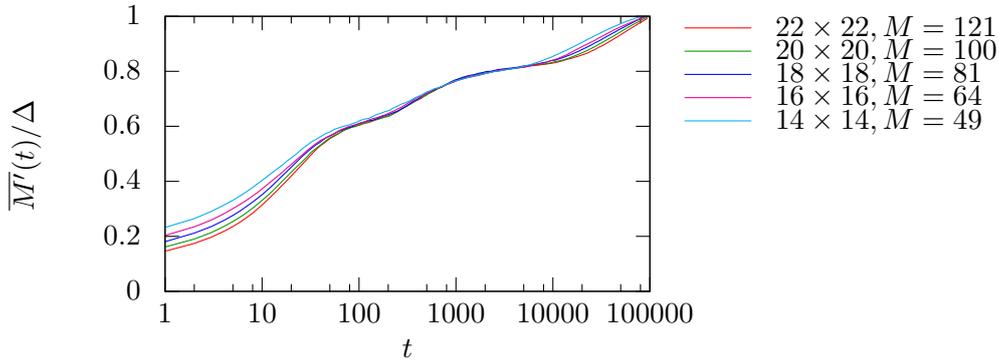


Figure 3.22.: Time evolution of the mass of the heaviest condensate on regular graphs. The averaged is computed over a sample of $\mathcal{N} = 10000$ $q = 4$ regular graphs. Note, that time is not rescaled in this plot, no scaling of condensation time can be seen here. However, the heaviest condensate mass grows over at least two plateaus of slower evolution. Although the condensation process is not completely recorded, the displayed cut-out shows most of it. Error bars have been omitted to provide an unobstructed view without plot symbols, however the biggest errors are in the order of 8×10^{-3} in regimes of stronger growth.

time series of the largest condensate mass rather exhibits plateaus of slower growth which are at similar times for all studied system sizes.

Another difference is that the condensate size, i.e. the number of sites it occupies is relatively small compared to the size of the condensate on a square lattice. After $t = 10^5$ MC sweeps the following average condensate sizes were observed:

N sites	M	condensate size
14×14	49	3.4
16×16	64	3.8
18×18	81	4.1
20×20	100	4.6

This might either indicate just that the condensate occupies only few sites or that the condensate further shrinks and occupies only a single site in the steady state. With the available data this cannot be finally decided in this thesis.

4. Monte Carlo Generation of Complex Networks

To numerically study the behavior of the described transport processes on arbitrary structures such as complex networks being able to sample graphs with a given set of prescribed structural properties is essential. As the sampling of graphs with such specific structural properties is interesting for many applications, the approach and procedure is covered in detail in this chapter. From the set of all graphs, such with desired properties will be sampled by means of Markov chain Monte Carlo methods while providing a sensible means of statistical weight calculation. At the end of the chapter stands an object oriented toolkit to perform Monte Carlo generation of equilibrated complex graphs and an estimation of diverse structural properties. It is used in this thesis as well as it is intended for further general use.

The science of complex networks is a relatively new field of research developing at the interface of physics, biology, social and computer sciences. It strives for a better understanding of the dynamics and general laws of creation and growth of networks as well as how processes on network topologies are affected by structure. There are numerous obvious, everyday examples of networks, such as traffic infrastructure, social contacts of persons, neuronal networks in nervous systems, communication networks or the worldwide web. Also many more kinds of networks that can be constructed from relationships or associations of things show properties of complex networks. Examples for such networks include science collaboration networks, where authors are connected if they have at least one shared publication [27], networks of protein conformations, with conformations being connected if they can be transformed into each other by a move from a given elementary move set [28] or in linguistics, networks of words with synonyms being connected. Some interesting problems involving networks as underlying structure include the spread of epidemics on social networks, the emergence of complex network properties due to growth and evolution of networks like the worldwide web or spread of opinions including interaction with networks in the form of rewiring of social contacts.

The basis of the theory of complex networks was established when Erdős and Rényi discovered the importance of probabilistic methods in the study of graphs [29,30]. Analysis of the structure of many existing networks led to the discovery of several properties that complex graphs frequently show. The most important and well known concepts are clustering, small worlds and scale-free behavior. Small worlds, for example, describe the existence of rather short paths between any two nodes of a graph as famously shown by Milgram [31]. Due to such features which will be partially discussed in Section 4.2, that are not found in random graphs, complex networks establish an important class of

4. Monte Carlo Generation of Complex Networks

random graphs. Several overviews of the topic with varying perspective have been published for example by Dorogovtsev et. al [3, 32], Albert et. al [33] and Waclaw [34].

This chapter is divided into five sections. After this introduction several definitions covering networks and graph elements are given, followed by a section on the typical structural quantities observed in real-world complex networks. A fourth section is dedicated to the description of statistical ensembles of networks and in the fifth section this is translated and applied to draw samples from such statistical ensembles employing Monte Carlo techniques. This is followed by the presentation of three examples of graph ensembles, that were tuned to specific properties and the corresponding simulated data. In the last section, a brief outlook is given. With the fundamentals provided in this chapter, the technical documentation of the toolkit is provided in the library package available from <http://www.physik.uni-leipzig.de/~nagel/graphgen>. Finally, a diagram of the workings of the provided modular toolkit and a minimal example simulation program is provided in the appendix B.

4.1. Definitions in the Context of Graphs

As the natural representation of real-world networks are graphs, a rather general definition shall be given to start with. A *directed graph* is a pair $G(V, E)$ consisting of a set of vertices V and a set of (directed) edges E , ordered pairs $\langle u, v \rangle \in E, u, v \in V$ connecting the vertex u to v , but not in the opposite direction. The number of vertices and edges in a graph is denoted by N and L respectively. The term node is commonly used as a synonym for vertex and the same goes for link and edge accordingly.

Additionally, there are *undirected graphs* consisting of nodes connected by undirected, symmetric edges. This can be achieved by different different definitions, where the most common differs from the undirected graph only in constituting the set of edges E to be a set of unordered pairs $\langle u, v \rangle$. However a more constructive definition seems more adequate in this context: A directed graph that contains an inverse directed edge $\langle v, u \rangle \in E$ for every edge $\langle u, v \rangle \in E$ is called *undirected graph*. This reproduces the representation of undirected graphs that will be used later on.

A useful generalization of each of the two graph concepts is the *pseudograph*, also called multigraph or degenerate graph. It is constructed from these by replacing the set of edges with a corresponding multiset of edges, thus allowing multiple identical edges. If a pair of nodes is connected by more than a single edge $\langle i, j \rangle$, the number of edges with the same endpoints shall be denoted as m_{ij} and called connectivity of the corresponding nodes. Edges $\langle i, i \rangle$ that connect to the same node as both endpoints are possible and referred to as unit loops. For distinction to pseudographs, graphs without multiple edges and unit loops are commonly called *simple graphs*. A simple graph can at most have $N(N - 1)/2$ edges, then being a *complete graph*.

Any directed graph or pseudograph and therefore network can be fully described by its adjacency matrix A , with entries A_{ij} yielding the number of edges

4.2. Characteristics of Complex Graphs

$\langle i, j \rangle$. From the above definition follows, that $A_{ij} = A_{ji}$ for undirected graphs. However, only such shall be considered here. There is also a number of properties that can directly be calculated from the adjacency matrix A . Although the exact node labels of a graph representation do not matter to the shape of a network, some distinction of the detail level of representation is useful. A graph represented with assigned node labels, as by its adjacency matrix, shall be called *labeled* graph and one with unique assigned edge labels a *fully labeled* graph. The latter, however, cannot be identified with its adjacency matrix and will be used only at the implementation of the sampling method. The network described by a representation is of course obtained by dropping any labels.

Starting with the most important local quantity, the *degree* q_i of a node characterizes the local structure just like coordination numbers do for periodic lattices. It is defined as the the number of edges incident with the corresponding node $q_i = \sum_j A_{ij}$. Hence, for a simple undirected graph the degree of a node is just the number of nearest neighbors. The average degree \bar{q} of a graph is then $\bar{q} = 1/N \sum_i q_i = 2L/N$ because every edge adds to two nodes degrees. With this, the *degree distribution*

$$\Pi(q) = \frac{1}{N} \sum_i^N \delta_{q_i, q} \quad (4.1)$$

is defined as the probability to draw a node with degree q randomly from the graph.

A *path* on a graph can be defined along a series of edges, each pair sharing one endpoint¹ and all intermediate nodes being distinct. A closed path, is called a cycle of order n if there are n intermediate edges along the path. If for every two nodes in a graph there exists a path between them, the graph is called connected. If it is not, it can be split into connected subgraphs consisting of distinct subsets of V and E as in Figs. 4.2(a) and (d). The number of cycles of order n can easily be obtained as the trace $1/2 \text{Tr} A^n$, as this matrix has the number of paths $i \rightarrow j$ in its elements (i, j) . The factor $1/2$ is a correction for undirected graphs to avoid double counting.

4.2. Characteristics of Complex Graphs

The features of complex networks mostly have been discovered in the empirical study of real world networks due to the vast availability of data including topological information of real networks. Important attributes are as stated, the scale-free behavior or power law degree distributions, small worlds and clustering. Interesting related properties are also degree-degree correlations, maximal degrees or the number of cycles in a graph which will be discussed in the following. It is instructive to compare complex networks to random graphs in terms of these quantities. When mentioning random graphs in the following, this refers to the Erdős-Renyi-model [29] which will be described in the next section and has been studied thoroughly since its introduction (1959). The

¹or the endpoint and origin respectively for directed graphs

4. Monte Carlo Generation of Complex Networks

quantities discussed below and the estimates of the corresponding ensemble averages can be calculated directly using the library provided.

The degree distribution $\Pi(q)$ is, as defined above, the probability of drawing a random node with degree q , therefore having exactly q nearest neighbors. As defined in (4.1) it is normalized $\sum_q \Pi(q) = 1$ and its mean value is therefore the average degree $\bar{q} = \sum_q q \Pi(q)$. Studies of the properties of real-world networks discovered that the degree distributions of many of them follow a *power-law*

$$\Pi(q) \propto q^{-\gamma} \quad (4.2)$$

for an interval of intermediate degrees $1 \ll q \ll N$ and typical exponents $2 < \gamma < 4$. At small degrees and at the cutoff of the degree distribution, characteristic behavior of specific network types is observed. This form of the degree distribution alone has been surprising as it is quite distinct from the expected form for random graphs, which is a binomial distribution with parameters $N - 1$ and the probability p that two nodes are connected. A related quantity is the *degree-degree correlation function* $\epsilon(k, q)$ which yields the probability to randomly draw an edge connecting nodes with degrees k, q . The resulting matrix is symmetric for undirected graphs, its upper triangle is normalized $\sum_{k < q} \epsilon(k, q) = 1$ and it incorporates the degree distribution of the graph by

$$\Pi(q) = \frac{\bar{q}}{q} \sum_k \epsilon(k, q) = \frac{\bar{q}}{q} \sum_k k \epsilon(q, k). \quad (4.3)$$

A more convenient definition of the same quantity is *average neighbor degree*

$$\bar{k}_{\text{nn}(q)} = \frac{\sum_{\langle i, j \rangle} q_i}{q_j \sum_{\langle i, j \rangle} 1}, \quad (4.4)$$

which gives the average degree of neighbors of nodes with the given degree q . It can also be calculated from the degree-degree-correlation function

$$\bar{k}_{\text{nn}(q)} = \frac{\bar{q}}{q \Pi(q)} \sum_k k \epsilon(k, q) \quad (4.5)$$

similar to the degree distribution. There is a related quantity describing the growth of the average neighbor degree with node degree called *assortativity*. A graph is assortative, if the average neighbor degree grows with node degree, and disassortative if it decreases. Assortativity can be expressed as a simple coefficient

$$\mathcal{A} = \frac{\text{Tr } \epsilon - \text{Tr } \epsilon_{\text{r}}}{1 - \text{Tr } \epsilon_{\text{r}}} \quad \begin{cases} < 0 & \text{disassortative} \\ = 0 & \text{uncorrelated} \\ > 0 & \text{assortative} \end{cases}, \quad (4.6)$$

by subtracting the factorized degree-degree probability of a graph with uncorrelated degrees but identical degree distribution

$$\epsilon_{\text{r}}(k, q) = \frac{1}{k} \Pi(k) q \Pi(q) \quad (4.7)$$

4.2. Characteristics of Complex Graphs

and normalizing the result. It was found that most social networks, like science collaboration graphs or networks of acquaintances are assortative. Disassortative degree-degree-correlations are common for real-world networks created for a specific purpose, like the internet or power grids [35]. In comparison, random graphs show uncorrelated degrees.

To understand the property of *small worlds* it is useful to look at paths between the nodes of a graph first. When two nodes i, j are connected, then there is at least one shortest path between them, the length of which is the distance $d(i, j)$ between them. The *diameter* of a graph is then the maximal distance of any two nodes in the graph

$$\text{diam } G = \max_{i, j \in V} d(i, j) . \quad (4.8)$$

If a graph is not connected, the diameter would strictly be infinite but is often defined as the maximum diameter of its connected components. An interesting related quantity is the average distance of any two nodes in the graphs. Networks exhibiting small average distances or small diameter with respect to their number of nodes are then called small worlds. In the experiment mentioned above, Milgram found a typical distance of 6 in the network of acquaintances among the approximately 300 million people in the USA. After a more specific definition, a network is a small world if the typical distance or diameter grows with the logarithm of the system. However, it was shown that complex networks share this property with random graphs.

Clustering describes the existence of groups of nodes where every member is connected with all other members. In terms of social networks, a cluster is a group of persons where two people are friends and if they have friends in common and vice versa. This means triangle constellations are common, giving a means to measure the clustering coefficient as the fraction of triangles in the number of connected triples

$$C = \frac{n_{\Delta}}{n_{\text{triples}}} , \quad (4.9)$$

where every three nodes that are connected by at least two edges are connected triples. A complete simple graph has then a clustering coefficient $C = 1$ due to every triple also being a triangle. Another measure is common to estimate the clustering coefficient locally at a given node i . This is again, the fraction of triangles that share that node in the number of triples that are there. However, the number of triples can be calculated more conveniently from the degree q_i giving

$$C_i = \frac{e_i}{q_i(q_i - 1)/2} . \quad (4.10)$$

The number of triangles is expressed by the number of edges e_i between the nearest neighbors of i . Both definitions are qualitatively and roughly numerical consistent when the average of C_i is taken for all nodes. The number of triangles may be calculated as

$$e_i = \frac{1}{2} (a^3)_{ii} \quad (4.11)$$

4. Monte Carlo Generation of Complex Networks

using the adjacency matrix. Clustering coefficients of real-world networks are in the order of $C \sim 0.6$ for science collaboration graphs for several different studied networks with sizes of 1×10^4 to 1.5×10^6 authors [27]. An example of a real-world network with relatively weak clustering of $c = 0.08$ is the power grid of the western united states with about $n \sim 5000$ nodes [36]. The clustering coefficient of random graphs is $c_{\text{rand}} \sim \bar{k}/n$ and quite distinct to most real-world networks.

4.3. Statistical Ensembles of Graphs

There are two intuitive approaches to describe and simulate the formation of networks: growing networks and equilibrated networks. In the former, the time evolution of the developing network is studied in terms of the influence of mechanisms creating the graph. In the latter approach — onto which this chapter is focused — one constructs a statistical ensemble of graphs and employs methods of statistical mechanics to study the characteristics of typical graphs.

The statistical ensemble is one of the most basic concepts in statistical physics. It is usually constructed from the phase space of a Hamiltonian system combined with the Gibbs measure identifying the probabilities of system states. The phase space is easily identified with the set of possible graphs but as networks are not Hamiltonian systems a statistical weight must be defined to obtain a corresponding probability measure.

At first the Erdős–Renyi model for random graphs is visited to construct an ensemble of equiprobable graphs. The Erdős–Renyi–model for random graphs $G(N, p)$ describes a method to construct simple graphs instead of randomly drawing such graphs from the set of all simple graphs with N nodes [29]. Starting with an empty graph containing only N nodes, every edge $\langle i, j \rangle$ possible in the graph is considered and added with fixed probability p independently of any other added edges. The related $G(N, L)$ model was introduced in the same publication and describes the construction of random networks with a fixed number of edges: A Pair of nodes is randomly chosen as endpoints for a new edge. It is added if the nodes are not linked with an edge, otherwise rejected. This is repeated until the graph contains L edges.

An equivalent of the latter method is to enumerate every possible $N \times N$ adjacency matrix A with elements $a_{ij} \in \{0, 1\}$, diagonal elements $a_{ii} = 0$ and the given number of edges $L = \sum_{i>j} a_{i,j}$ and pick one at random. Hence, with this method every possible labeled graph is constructed with the same probability. Figure 4.1 shows the possible labeled graphs for $N = 4$ nodes and $L = 2$ edges representing two network shapes. The probabilities of the networks is then $p_A = 4/5$ and $p_B = 1/5$ respectively. Evidently different networks are not equiprobably constructed, only graphs are. Because the statistical weights $w(A), w(B)$ of the networks A and B are proportional to the probabilities of their appearance p_A, p_B , only the proportionality constant, being the weight of a single labeled graph remains unset. For convenience this is chosen $1/N!$, which compensates the factor from permutations of node labels.

The partition function $Z(N, L)$ for networks constructed from the Erdős–Renyi

4.3. Statistical Ensembles of Graphs

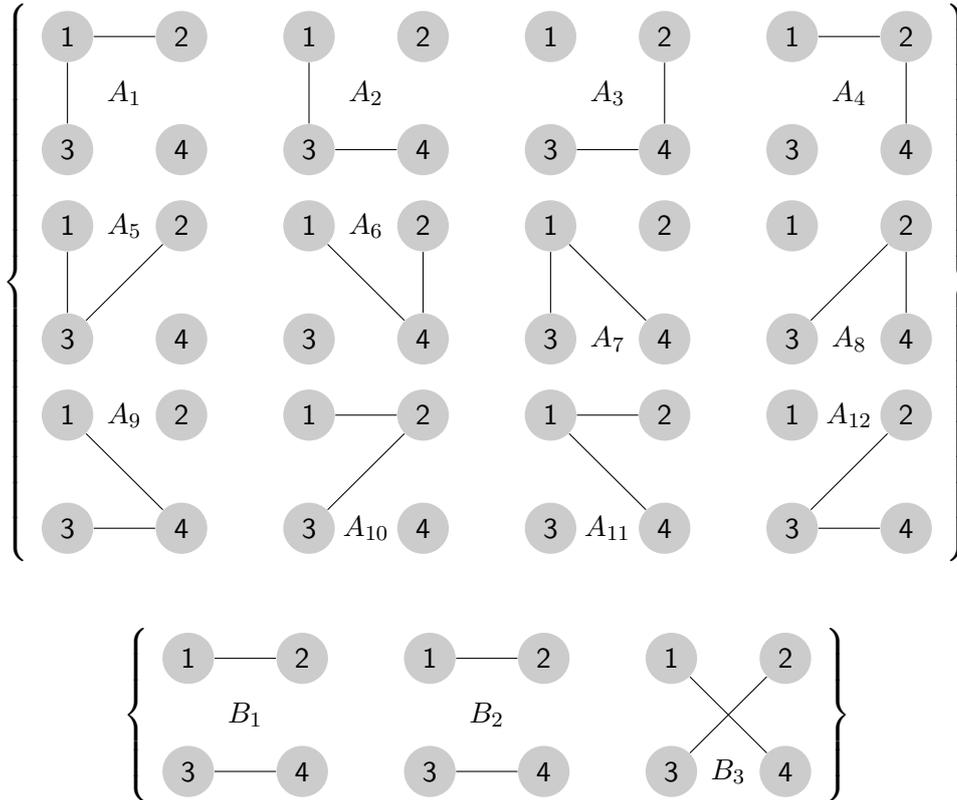


Figure 4.1.: All possible simple labeled graphs with $N = 4$ nodes and $L = 2$ edges are shown. There are only two distinct networks A and B possible represented by the labeled graphs in the upper or lower brackets. The networks A and B have $n_A = 12$ and $n_B = 3$ different representations as labeled graphs. One could also label edges, and thus use fully labeled graphs as network representations. This does not change adjacency matrices and thus only adds a factor $1/L!$ to the statistical weight of each graph.

4. Monte Carlo Generation of Complex Networks

ensemble is then

$$Z(N, L) = \sum_{\alpha' \in G'(N, K)} = \sum_{\alpha \in G(N, L)} w(\alpha), \quad (4.12)$$

with the set of labeled graphs $G'(N, L)$, the statistical weight $w(\alpha) = n_\alpha/N!$, where n_α is the number of labeled graphs α' for the network α . The expectation value of a given property \mathcal{O} is then defined as usual per

$$\langle \mathcal{O} \rangle = \frac{1}{Z(N, L)} \sum_{\alpha \in G(N, L)} w(\alpha) \mathcal{O}(\alpha). \quad (4.13)$$

Thus the properties of the “typical” network in a given ensemble can be studied. However, to study more interesting ensembles of graphs, where typical graphs have different features of real-world networks, the defined fundamental weight $1/N!$ for labeled graphs is not sufficient. This can be accomplished by introducing an additional functional weight which in general is different for each network. By adjusting this functional weight, the properties of typical networks can be tuned towards the desired features.

In principle the form of the weight is arbitrary and can depend on any topological information a graph offers. However, some general factorizing forms are convenient and efficient especially for Monte Carlo sampling. The most simple form is the node degree weight

$$w(\alpha) = \prod_i g(q_i), \quad (4.14)$$

with a function $g(q)$ of the nodes degree giving the local weight factor. This approach is already usable and for example an ensemble of graphs with

$$g(q) = \frac{4q!}{q(q+1)(q+2)} \quad (4.15)$$

reproduces the degree distribution of Barabási–Albert graphs [33] quite well. To introduce degree-degree correlations between nodes, a two-point function $g(q_i, q_j)$ of degrees of endpoints of an edge $\langle i, j \rangle$ is used to define the edge weight

$$W(\alpha) = \prod_{\langle i, j \rangle} g(k_i, k_j). \quad (4.16)$$

Again, the local weight function $g(\cdot, \cdot)$ defining the statistical weight of graphs may be chosen arbitrarily. Of course there are many ways of defining functional weights which might be appropriate for some cases, but these are not the scope of this chapter. Also, when applying Markov chain sampling to these ensembles it is a major advantage to employ products of local weight functions as statistical weights, as otherwise the total graph weight had to be computed at every step anew.

Until now, only ensembles of graphs with fixed size and number of edges were considered. In the following, similar ensembles are described briefly, for a detailed

4.4. Monte Carlo Generation of Equilibrated Networks

introduction refer to [34]. For a distinction of these, an ensemble with fixed N, L will be referred to as a *canonical ensemble* in analogy to the conservation of particles in the canonical ensemble of particles in a thermodynamic system. The *grandcanonical ensemble* refers to ensembles of graphs with fixed size N but varying number of edges. Without a functional weight this is equivalent to the ensemble constructed from the $G(N, p)$ model. The partition function is

$$Z(N, \mu) = \sum_L \exp(-\mu L) Z(N, L), \quad (4.17)$$

with fugacity $p/(1-p) = \exp(-\mu)$ in the corresponding Erdős–Rényi ensemble. The weight is then $w(\alpha) = \exp(-\mu L(\mu))/N!$ where μ tunes the potential to add or remove edges similar to the chemical potential in Hamiltonian systems.

Additionally in analogy to the concept of equiprobable microstates with identical energy in Hamiltonian systems, a *microcanonical ensemble* can be defined. In the case of a node degree weight type functional weight (4.14), this is accomplished by fixing the sequence of node degrees (q_1, q_2, \dots, q_N) and the canonical ensemble is constructed by taking the sum over all degree sequences with fixed number of edges. However the analogy of equiprobable microstates is broken as soon as introducing a less local weight function such as the edge weight (4.16). Nevertheless it is useful to compute properties of a graph ensemble with fixed degree sequence for example to obtain disordered networks or lattices.

4.4. Monte Carlo Generation of Equilibrated Networks

This section specifically covers the generation of complex graphs using the ensembles discussed above and also including some technical discussion of the choices in realization of the library.

First of all the former graphgen library [37] must be credited as the basis of this implementation. It offered graph representation, generation of equilibrated graphs using vertex weights and the calculation of some graph properties like the clustering coefficient and degree distributions. Also a similar rewrite of the library has been started by P. Białas, co-author of the original graphgen². The main goals for the successor library were the ability to sample edge weighed graphs as well as vertex weighed graphs and sampling from adjustably restricted graph subsets in a modular, object oriented approach to ease extensibility. It can be obtained from <http://physik.uni-leipzig.de/~nagel/graphgen/>.

It is useful to be able to restrict the shapes of sampled networks in several ways. The following classes in the sense of subensembles will be considered:

- (i) *Trees*, which are graphs without any closed path along edges,
- (ii) *simple graphs* contain only single undirected edges and no unit loops,
- (iii) *pseudographs without unit loops* may contain multiple edges between pairs of nodes

²At the state of the library I continued this task it was mostly the basis of graph representation code.

4. Monte Carlo Generation of Complex Networks

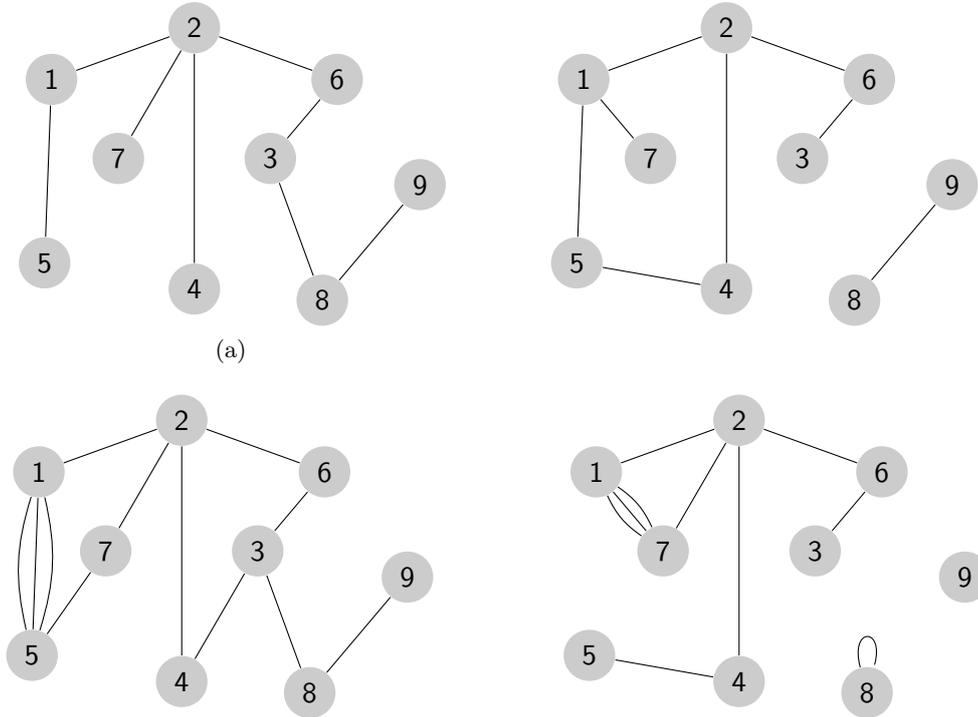


Figure 4.2.: Examples of undirected graphs to illustrate the distinction of graph classes. (a) tree graph, (b) simple graph, (c) pseudo graph without unit loops, (d) pseudo graph with unit loops. The graphs in (a) and (d) are disconnected as they contain separate subgraphs.

(iv) and of course arbitrary undirected *pseudographs* without any other restrictions.

Figure 4.2 is provided for a visual distinction of these graph classes.

4.4.1. Graph representation

Although labeled graphs are identified by their adjacency matrices, this is not necessarily the best choice of representation in the process of graph generation or calculation of quantities. The adjacency list representation scheme will be described followed by a brief discussion of advantages and disadvantages both schemes yield.

As mentioned, the $N \times N$ adjacency matrix uniquely describes a labeled graph by the number of directed edges $\langle i, j \rangle$ as elements a_{ij} . As an example, the graph in Figure 4.3 corresponds to

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}.$$

4.4. Monte Carlo Generation of Equilibrated Networks

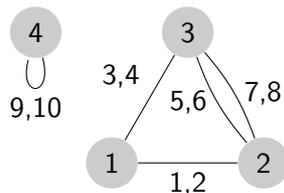


Figure 4.3.: An example of an undirected graph as an illustration to different representation methods discussed in Section 4.4.1. Nodes and pairs of edges are fully labeled for convenience and reference in the representations.

It yields advantages in fast lookup and modification of edges connecting node pairs which are $O(1)$ operation. Enumeration of neighbors, needed to calculate local edge weights or non-local quantities like the clustering coefficient instead is $O(N)$, as a complete column of the adjacency matrix must be evaluated. Another disadvantage is the square growth with the number of sites when generating sparse graphs.

The adjacency list scheme represents the graph structure as lists of emerging directed edges for every node along with their destination. Table Tab. 4.1 explains the used representation of the graph in Figure 4.3. In this representation, operations like modifying and looking up edge connectivity as well as finding a node's neighbors roughly scale with the average degree as a node's list of edges is evaluated. For sparse graphs the amount of memory needed only scales linearly and thus it is possible to generate larger networks than with using adjacency matrices.

4.4.2. Local updates and transition rates

The Metropolis algorithm is used with a local update set to sample graphs. The probability accepting a proposed update

$$P(\alpha \rightarrow \beta) = \min \left\{ 1, \frac{w(\beta)}{w(\alpha)} \right\}$$

according to (2.5) depends now only on the chosen ensemble and functional weight. Along with a description of the used local updates the rates will be calculated in the following sections. An overview of the updates is given in Figure 4.4. To realize the sampling of a specific graph class, such as simple graphs, only updates that comply with the given restriction are proposed. For example, to sample only simple graphs, one must check if the new configuration creates a unit loop or multiple links. A detailed overview of the rules for update proposal is given in Table 4.2.

When calculating the acceptance rates for ensembles with functional weights constructed from local degree or edge weights the ratios

$$r(q) = \frac{g(q+1)}{g(q)} \quad \text{and} \quad r(q_i, q_j) = \frac{g(q_i+1, q_j)}{g(q_i, q_j)}$$

4. Monte Carlo Generation of Complex Networks

Table 4.1.: Adjacency list representation of the graph in Fig. 4.3. The adjacency lists are encoded on the right hand side of the table as follows. The first edge leaving column refers to the first element of the adjacency list of each node, while the next edge column links list items. The target node refers to the endpoint of each edge in the list. To speed up calculation of update probabilities, the degree of each node is also stored.

node index	degree	first edge	edge index	next edge	target node
1	2	3	1	-	2
2	3	7	2	-	1
3	3	8	3	1	3
4	2	9	4	-	1
(a) nodes			5	2	3
			6	4	2
			7	5	3
			8	6	2
			9	10	4
			10	-	4
			(b) edges		

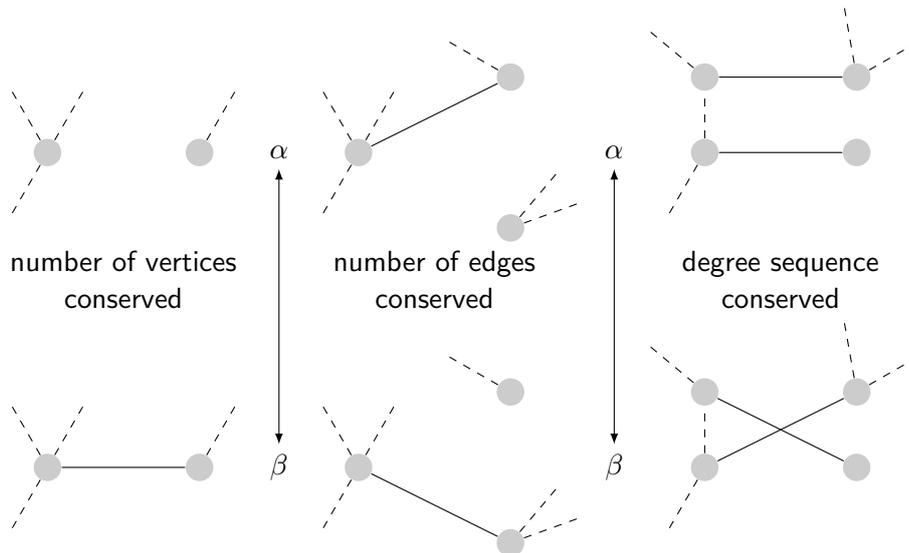


Figure 4.4.: Overview of the fundamental local graph update mechanisms. From left to right the update mechanisms are less generic, thus conserving more properties and yielding smaller network ensembles.

4.4. Monte Carlo Generation of Equilibrated Networks

Table 4.2.: Details of update proposition for implemented combinations of graph shapes as in Figure 4.2 and graph ensembles. The generation of trees using the grandcanonic ensemble is not implemented as connected trees are split up. Note that the constraint $\langle i, j \rangle \neq \langle h, l \rangle$ means that endpoint nodes are mutually distinct.

microcanonic ensemble	canonic ensemble	grandcanonic ensemble
<i>connected trees</i>		
draw two leaf nodes $h \neq k$ with $q_h = 1 = q_k$ at random without replacement, their single leaving edges are then $\langle h, j \rangle$ and $\langle k, l \rangle$ respectively	draw a leaf node h with $q_{h=1}$ at random, its single leaving edge is $\langle h, k \rangle$ draw target vertex $l \neq k, h$ at random	this combination breaks up connected trees
<i>simple graphs</i>		
draw two edges $\langle h, j \rangle \neq \langle k, l \rangle$ with $j \neq k, h \neq l$ and $m_{hl} = 0 = m_{jk}$ at random without replacement	draw an edge $\langle h, k \rangle$ at random, draw a random target node $l \neq h, k$, $m_{hl} = 0$	draw an edge $\langle h, k \rangle$ at random for removal or draw two nodes $h \neq k$ with $m_{hk} = 0$ at random without replacement to connect
<i>pseudographs, no unit loops</i>		
draw two edges $\langle h, j \rangle, \langle k, l \rangle, j \neq k, h \neq l$ without replacement	draw an edge $\langle h, k \rangle$ and a node $l \neq h, k$ at random	draw an edge $\langle h, k \rangle$ at random to delete or draw two nodes $h \neq k$ to connect at random
<i>pseudographs</i>		
draw two edges $\langle h, j \rangle \neq \langle k, l \rangle$ at random without replacement	draw an edge $\langle h, k \rangle$ and a node $l \neq k$ at random	draw an edge $\langle h, k \rangle$ to delete or two nodes h, k to connect at random with replacement

4. Monte Carlo Generation of Complex Networks

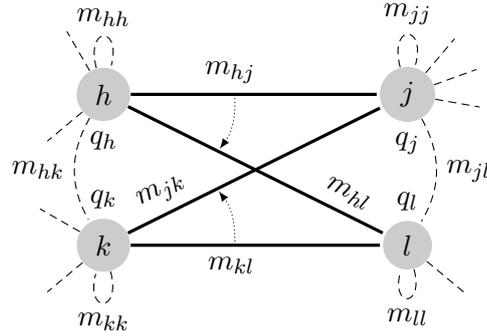


Figure 4.5.: Connectivity details for the *micro-canonical* update mechanism. Thick edges are cross switched as indicated. Dashed edges are not subject to connectivity changes during update, so their weight remains.

are useful to shorten notation. Because only undirected graphs are sampled the local edge weight function is symmetric $g(q_i, q_j) = g(q_j, q_i)$. The product over any undirected edge $\langle i, j \rangle$ is indicated by $\prod_{\langle i, j \rangle}$, while $\prod_{h, i}$ stands for the product over all directed edges emerging a given node h with endpoints i being neighbors of h . The number of edges $\langle i, j \rangle$ in pseudographs is labeled m_{ij} . For a shorter notation of total graph weight ratios, $W(\gamma)$ shall denote the total weight of the biggest subgraph unaffected by the proposed update.

4.4.3. The micro-canonical network ensemble

In the micro-canonical network ensemble the update process must conserve the degrees of every node in the graph. As shown in Figure 4.5 this is achieved by “crossing” a pair of links. In detail, two directed links are picked at random from the graph. If the update is accepted, their targets are swapped as well as the origin nodes of their inverse links. It is easy to figure that node degrees are also conserved in the special cases where the selected links share one or two nodes as endpoints. As a result only the degree-degree-pairing of the links changes the graphs structure. I start with the common vertex weight method.

To show the ergodicity of this update the following strategy is used to construct any graph configuration from a given graph by consecutive updates. First, reduce as much edges per node as possible to unit loops, by crossing edges. The result is that only nodes with odd degree ($q_i = 2n + 1$) will have a remaining outbound edge to another such node. These remaining edges may then be rearranged followed by the reconnection of the edges reduced to unit loops to meet the new configuration. However, this argument apparently works only for pseudo graphs with unit loops.

Vertex weighed graphs

The micro-canonical ensemble is designed to consist of equiprobable graphs when the graph weight is constructed using local degree weights as in (4.14).

4.4. Monte Carlo Generation of Equilibrated Networks

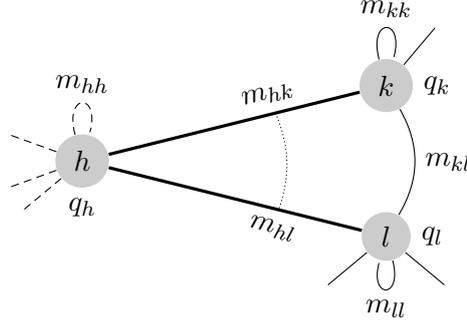


Figure 4.6.: Details for the *canonical* update mechanism. The thick edge is switched from k to l as indicated. As before, the endpoint connectivity of dashed edges is not changed.

Updates are accepted always as is

$$\frac{W(\beta)}{W(\alpha)} = 1. \quad (4.18)$$

Edge weighed graphs

The situation of equiprobable graphs changes when going to edge weights defining the total weight of sampled graphs. From Figure 4.5 it can be seen that the only changes in connectivity are $m_{kl} \rightarrow m_{kl} - 1$, $m_{hj} \rightarrow m_{hj} - 1$ and $m_{hl} \rightarrow m_{hl} + 1$, $m_{kj} \rightarrow m_{kj} + 1$ for a given proposed update. For convenience the weight of the unaffected part of the graph is separated as $W(\gamma)$. With this, the partial weight representing only edges with changing connectivity is written as $W(\alpha)/W(\gamma)$ or $W(\beta)/W(\gamma)$ respectively after the proposed update. With

$$\frac{W(\alpha)}{W(\gamma)} = g(q_h, q_j)^{m_{hj}} g(q_k, q_l)^{m_{kl}} g(q_h, q_l)^{m_{hl}} g(q_k, q_j)^{m_{kj}} \quad (4.19)$$

and

$$\frac{W(\beta)}{W(\gamma)} = g(q_h, q_j)^{m_{hj}-1} g(q_k, q_l)^{m_{kl}-1} g(q_h, q_l)^{m_{hl}+1} g(q_k, q_j)^{m_{kj}+1}, \quad (4.20)$$

the weight ratio reads

$$\frac{W(\beta)}{W(\alpha)} = \frac{g(q_h, q_l)g(q_k, q_j)}{g(q_h, q_j)g(q_k, q_l)}. \quad (4.21)$$

It should be noted that — different to the canonical and grand-canonical ensembles — this weight ratio also holds for special cases with pseudographs like multiple edges and unit loops.

4.4.4. The canonical network ensemble

In this graph ensemble the number of links must be conserved. This is achieved by picking a random link from the graph and replacing one of its ending points

4. Monte Carlo Generation of Complex Networks

with a randomly selected node as indicated in Figure 4.6. This changes the degree of the endpoint node's before (k) and after (l) the proposed update. To understand that this updates is ergodic, one can figure, that any edge $\langle h, k \rangle$ can be moved to two distinct endpoints $\langle m, l \rangle$ by two updates via the intermediate step $\langle h, l \rangle$.

Vertex weighed graphs

With the weight of the unaffected subgraph

$$W(\gamma) = \prod_{\substack{\langle i,j \rangle \\ i,j \notin \{k,j\}}} g(q_i, q_j)$$

the weight of the changed subgraph is given by the two nodes with changing degree and reads before the update $W(\alpha)/W(\gamma) = g(q_k)g(q_l)$ and after the update it is $W(\beta)/W(\gamma) = g(q_k - 1)g(q_l + 1)$ because the edge no longer connects to k but to l now and degrees change. The resulting acceptance rate reads then

$$\frac{W(\beta)}{W(\alpha)} = \frac{g(q_k - 1)g(q_l + 1)}{g(q_k)g(q_l)} = \frac{r(q_l)}{r(q_k - 1)}. \quad (4.22)$$

Edge weighed graphs

The local part of the total graph weight can be calculated as a product over all edges emerging from the affected vertices h, k and l respectively. Because edges shared between two of these involved nodes are counted twice in this scheme, the corresponding squared weight factors must be canceled by correction factors. Correction factors also appear in special cases, when the selected edge is or is proposed to become a unit loop. In the calculation, the following cases are considered

- (a) h, k, l distinct, the most common case and also the only possibility for the sampling of simple graphs,
- (b) $h = k, h \neq l$, the selected edge is an unit $\langle h, h \rangle$ and becomes an edge $\langle h, l \rangle$,
- (c) $h \neq k, h = l$, the link $\langle h, k \rangle$ becomes a unit loop $\langle h, h \rangle$,
- (d) $k = l$ is a pathologic case as the proposed update would be just an identity operation and can therefore always be accepted and is not considered.

In the following h, k, l are assumed to be distinct if not stated otherwise. The weight of the unaffected subgraph is

$$W(\gamma) = \prod_{\substack{\langle i,j \rangle \\ i,j \notin \{h,k,l\}}} g(q_i, q_j).$$

4.4. Monte Carlo Generation of Equilibrated Networks

With this and Figure 4.6 the needed correction factors for the weight ratio before the update can be written as

$$\left(\frac{W(\alpha)}{W(\gamma)}\right) = \frac{\prod_{k,j} g(q_k, q_j) \prod_{l,j} g(q_l, q_j)}{g(q_k, q_k)^{m_{kk}/2} g(q_k, q_l)^{m_{kl}} g(q_l, q_l)^{m_{ll}/2}}. \quad (4.23)$$

The correction factor $1/g(q_k, q_k)^{m_{kk}/2}$ for example cancels the additional weight factors that might result from counting both directed edges of unit loops $\langle k, k \rangle$. Similarly the weights of edges $\langle k, l \rangle$ and possible unit loops $\langle l, l \rangle$ are canceled out.

The local weight part after the update needs correction factors as above and additional corrections because the products over neighbors are not aware the update changed the connectivity. With h, k, l distinct the local weight part is

$$\begin{aligned} \left(\frac{W(\beta)}{W(\gamma)}\right) &= \prod_{k,j} g(q_k - 1, q_j) \prod_{l,j} g(q_l + 1, q_j) \left[\frac{g(q_h, q_l + 1)}{g(q_h, q_k - 1)} \right] \\ &\cdot \left[\frac{g(q_k - 1, q_k - 1)}{g(q_k - 1, q_k)^2} \right]^{m_{kk}/2} \left[\frac{g(q_l + 1, q_l + 1)}{g(q_l + 1, q_l)^2} \right]^{m_{ll}/2} \\ &\cdot \left[\frac{g(q_k - 1, q_l + 1)}{g(q_k - 1, q_l)g(q_l + 1, q_k)} \right]^{m_{kl}}, \end{aligned} \quad (4.24a)$$

where the third term cancels the factor from the edge $\langle h, k \rangle$, and adds that for the switched edge $\langle h, l \rangle$. The fourth and fifth term correct factors with invalid endpoint degrees from unit loops $\langle k, k \rangle$ and $\langle l, l \rangle$ and finally the last term corrects weight factors with wrong node degrees for edges $\langle k, l \rangle$.

With similar corrections the weight parts in the second and third cases read

$$\begin{aligned} \left(\frac{W(\beta)}{W(\gamma)}\right)_{(h=k)} &= \prod_{h,j} g(q_h - 1, q_j) \prod_{l,j} g(q_l + 1, q_j) \left[\frac{g(q_h - 1, q_l + 1)}{g(q_h - 1, q_h - 1)} \right] \\ &\cdot \left[\frac{g(q_h - 1, q_h - 1)}{g(q_h - 1, q_h)^2} \right]^{m_{hh}/2} \left[\frac{g(q_l + 1, q_l + 1)}{g(q_l + 1, q_l)^2} \right]^{m_{ll}/2} \\ &\cdot \left[\frac{g(q_l + 1, q_h - 2)}{g(q_h - 1, q_l)g(q_l + 1, q_h)} \right]^{m_{hl}} \end{aligned} \quad (4.24b)$$

and

$$\begin{aligned} \left(\frac{W(\beta)}{W(\gamma)}\right)_{(h=l)} &= \prod_{h,j} g(q_h + 1, q_j) \prod_{k,j} g(q_k - 1, q_j) \left[\frac{g(q_k + 1, q_h + 1)}{g(q_k - 1, q_h + 1)} \right] \\ &\cdot \left[\frac{g(q_h + 1, q_h + 1)}{g(q_h + 1, q_h)^2} \right]^{m_{hh}/2} \left[\frac{g(q_k - 1, q_k - 1)}{g(q_k - 1, q_k)^2} \right]^{m_{kk}/2} \\ &\cdot \left[\frac{g(q_k - 1, q_k + 1)}{g(q_k - 1, q_h)g(q_h + 1, q_k)} \right]^{m_{hk}}. \end{aligned} \quad (4.24c)$$

By comparing the structure of the calculated weight parts after the suggested update a general form

$$\frac{W(\beta)}{W(\gamma)} = P_{\text{NN}} P_{\text{SL}} P_{\text{loop}} P_{\text{multiple}} \quad (4.25)$$

4. Monte Carlo Generation of Complex Networks

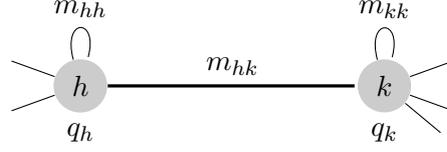


Figure 4.7.: Connectivity details for the *grand-canonical* update mechanism. The thick edge is either removed or added. As before the endpoint connectivity of dashed edges is not changed in the update.

can be written, where P_{NN} equals the two edge weight products for the nodes k and l , P_{multiple} is a correction for multiple edges $\langle k, l \rangle$ in pseudographs, P_{loop} corrects double counted unit loops and finally P_{SL} replaces the weight of the removed edge $\langle h, k \rangle$ with that of the now switched link $\langle h, l \rangle$. The considered special cases differ now only in the term P_{SL} which is

$$P_{\text{SL}} = \begin{cases} \frac{g(q_h, q_l + 1)}{g(q_h, q_k - 1)} & (h \neq k \neq l) \\ \frac{g(q_h - 1, q_l + 1)}{g(q_h - 1, q_h - 1)} & (h = k) \\ \frac{g(q_h + 1, q_h + 1)}{g(q_k - 1, q_h + 1)} & (h = l) \end{cases} . \quad (4.26)$$

The general form of the weight ratio for the canonical ensemble then calculates

$$\frac{W(\beta)}{W(\alpha)} = \frac{\prod_{l,j} r(q_l, q_j)}{\prod_{k,j} r(q_k - 1, q_j)} P_{\text{SL}} \left[\frac{r(q_k - 1, q_l)}{r(q_k - 1, q_l + 1)} \right]^{m_{kl}} \quad (4.27)$$

$$\cdot \left[\frac{r(q_k - 1, q_k)}{r(q_k - 1, q_k - 1)} \right]^{m_{kk}/2} \left[\frac{r(q_l, q_l + 1)}{r(q_l, q_l)} \right]^{m_{ll}/2} .$$

The structure of the calculated acceptance rate has the advantage of being simplified easily in cases where unit loops or multiple edges are forbidden.

4.4.5. The grand-canonical network ensemble

In this ensemble, two complementary updates are needed to sample the configuration space: addition and removal of edges, for details, see Figure 4.7. Hence, acceptance rates are calculated for either of them in this section. Also, as the number of links fluctuates, the fundamental weight of labeled graphs cannot be abandoned to calculate acceptance rates. It is shown in [38] that the acceptance rate for adding an edge is then

$$P_{\text{acc, add}}(\alpha \rightarrow \beta) = \min \left\{ 1, \frac{\exp(-\mu)N^2}{2(L+1)} \left(\frac{W(\beta)}{W(\alpha)} \right)^+ \right\} , \quad (4.28)$$

while the rate for removing a link reads

$$P_{\text{acc, remove}}(\alpha \rightarrow \beta) = \min \left\{ 1, \frac{\exp(\mu)2L}{N^2} \left(\frac{W(\beta)}{W(\alpha)} \right)^- \right\} , \quad (4.29)$$

4.4. Monte Carlo Generation of Equilibrated Networks

where N and L refer to the number of nodes and links before the update occurs. It is easy to see, that this update is ergodic, as any configuration in $N + N'$ can be reached by removing all N existing links and adding N' links to reach the new configuration.

Vertex weighed graphs

The weight ratios corresponding to edge addition and removal are calculated to be inserted in the form (4.28) and (4.29) respectively and as before a special case can emerge, when a unit loop is added or removed from the graph. The graph weight ratios for adding a link read

$$\left(\frac{W(\beta)}{W(\alpha)}\right)^+ = \frac{g(q_k + 1)g(q_l + 1)}{g(q_k)g(q_l)} = r(q_k)r(q_l) \quad (4.30a)$$

and

$$\left(\frac{W(\beta)}{W(\alpha)}\right)_{k=l}^+ = \frac{g(q + 2)}{g(q)} \quad (4.30b)$$

respectively. Quite similar, the ratios for link removal are

$$\left(\frac{W(\beta)}{W(\alpha)}\right)^- = \frac{1}{r(q_k - 1)r(q_l - 1)} \quad (4.31a)$$

and

$$\left(\frac{W(\beta)}{W(\alpha)}\right)_{k=l}^- = \frac{g(q - 2)}{g(q)} \quad (4.31b)$$

respectively.

Edge weighed graphs

The structure of the acceptance rates (4.28) and (4.29) is correct in this ensemble too and only the weight ratios remain. The calculation is similar to that in the canonical ensemble, with similar correction factors for edges count twice and a special case for adding or removing a unit loop from the graph. With

$$\frac{W(\alpha)}{W(\gamma)} = \frac{\prod_{k,j} g(q_k, q_j) \prod_{l,j} g(q_l, q_j)}{g(q_k, q_k)^{m_{kk}/2} g(q_k, q_l)^{m_{kl}} g(q_l, q_l)^{m_{ll}/2}} \quad (4.32)$$

and

$$\begin{aligned} \frac{W(\beta)}{W(\gamma)} = & \prod_{k,j} g(q_k + 1, q_j) \frac{g(q_k + 1, q_k + 1)^{m_{kk}/2}}{g(q_k + 1, q_k)^{m_{kk}}} \\ & \cdot \prod_{l,j} g(q_l + 1, q_j) \frac{g(q_l + 1, q_l + 1)^{m_{ll}/2}}{g(q_l + 1, q_l)^{m_{ll}}} \\ & \cdot \left[\frac{g(q_k + 1, q_l + 1)}{g(q_l + 1, q_k)g(q_{k+1}, q_l)} \right]^{m_{kl}} g(q_k + 1, q_l) , \end{aligned} \quad (4.33)$$

4. Monte Carlo Generation of Complex Networks

where factors for multiple, already existing edges $\langle k, l \rangle$ and unit loops $\langle k, k \rangle, \langle l, l \rangle$ are canceled and the last term accounts for the new link, the weight ratio is calculated as

$$\begin{aligned} \left(\frac{W(\beta)}{W(\alpha)} \right)^+ &= \prod_{k,j} r(q_k, q_j) \prod_{l,j} r(q_l, q_j) \cdot g(q_k + 1, q_l + 1) \\ &\cdot \left[\frac{r(q_k, q_l + 1)}{r(q_k, q_l)} \right]^{m_{kl}} \left[\frac{r(q_k, q_k + 1)}{r(q_k, q_k)} \right]^{m_{kk}/2} \left[\frac{r(q_l, q_l + 1)}{r(q_l, q_l)} \right]^{m_{ll}/2}, \end{aligned} \quad (4.34a)$$

joining correction factors and using the symmetry of the local weight $g(\cdot, \cdot)$ to shorten notation. In the case of an added unit loop where $q_k + 1$ is replaced by $q_k + 2$ and $k = l$ the ratio simplifies to

$$\begin{aligned} \left(\frac{W(\beta)}{W(\alpha)} \right)_{k=l}^+ &= \prod_{k,j} \frac{g(q_k + 2, q_j)}{g(q_k, q_j)} \left[\frac{g(q_k + 2, q_k + 2)g(q_k, q_k)}{g(q_k + 2, q_k)^2} \right]^{m_{kk}/2} \\ &\cdot g(q_k + 2, q_k + 2), \end{aligned} \quad (4.34b)$$

The ratio for link removal is calculated similar and reads in reads

$$\begin{aligned} \left(\frac{W(\beta)}{W(\alpha)} \right)^- &= \prod_{k,j} \frac{1}{r(q_k - 1, q_j)} \prod_{l,j} \frac{1}{r(q_l - 1, q_j)} \frac{1}{g(q_k - 1, q_l - 1)} \\ &\cdot \left[\frac{g(q_k - 1, q_l - 1)g(q_k, q_l)}{g(q_k - 1, q_l)g(q_l - 1, q_k)} \right]^{m_{kl}} \\ &\cdot \left[\frac{r(q_k - 1, q_k)}{r(q_k - 1, q_k - 1)} \right]^{m_{kk}/2} \left[\frac{r(q_l - 1, q_l)}{g(q_l - 1, q_l - 1)} \right]^{m_{ll}/2} \end{aligned} \quad (4.35a)$$

in the case of $k \neq l$. As for link addition, the corresponding weight ratio to removing a unit loop calculates to

$$\begin{aligned} \left(\frac{W(\beta)}{W(\alpha)} \right)_{k=l}^- &= \prod_{k,j} \frac{g(q_k - 2, q_j)}{g(q_k, q_j)} \left[\frac{g(q_k - 2, q_k - 2)g(q_k, q_k)}{g(q_k - 2, q_k)^2} \right]^{m_{kk}/2} \\ &\cdot g(q_k - 2, q_k - 2). \end{aligned} \quad (4.35b)$$

4.5. Exemplary Graph Ensembles and Properties

In this section I shall give three examples of graph ensembles with the corresponding properties estimated by the application of the discussed sampling methods. The ensemble in the first example is chosen to reproduce the degree distribution of the well-known Barabási–Albert model [33] of growing networks. This can be accomplished by using the local node degree function (4.15). Some of its properties are displayed in Figure 4.8.

In the second example edges between nodes of high and such of low degree shall be favored to produce mostly disassortative networks. A possible realization is the construction of a functional weight of form (4.16) using the local edge weight

$$g(k, q) = |q - k|^2. \quad (4.36)$$

The estimated properties displayed in Figure 4.9 clearly show that the typical networks have the shape of star graphs.

A third example ensemble is defined to prefer graphs with edges connecting nodes that have approximately the same degree. The local edge weight

$$g(k, q) = \frac{1}{1 + |q - k|^2} \quad (4.37)$$

is used to define the functional weight and the corresponding ensemble. A typical sample featuring the desired shape and estimated properties is displayed in Figure 4.10.

4.6. Further Remarks

The discussed methods to sample different types of networks employing statistical ensembles of graphs are provided in the created toolkit as well as methods for estimation of ensemble averages of the discussed quantities. Due to its the modular design, it is quite extensible. It is possible to supplement more general functional weights, sampling methods or define desired graph types by adding methods generating update propositions accordingly. For example a functional weight constructed of factors that are calculated from the whole neighborhood of every single node might be interesting. In principle it is even possible to use a functional weight, that is not factorized of local graph elements. However, depending on the specific weight, this would probably slow down the graph generation enormously. Another option is to implement more sophisticated simulation techniques by adding a module accordingly. For example multicanonical methods can be used to sample graphs [39], and might be supplemented. A rough graphical description of the modular structure of the toolkit and a brief example of using it is given in Appendix B.

4. Monte Carlo Generation of Complex Networks

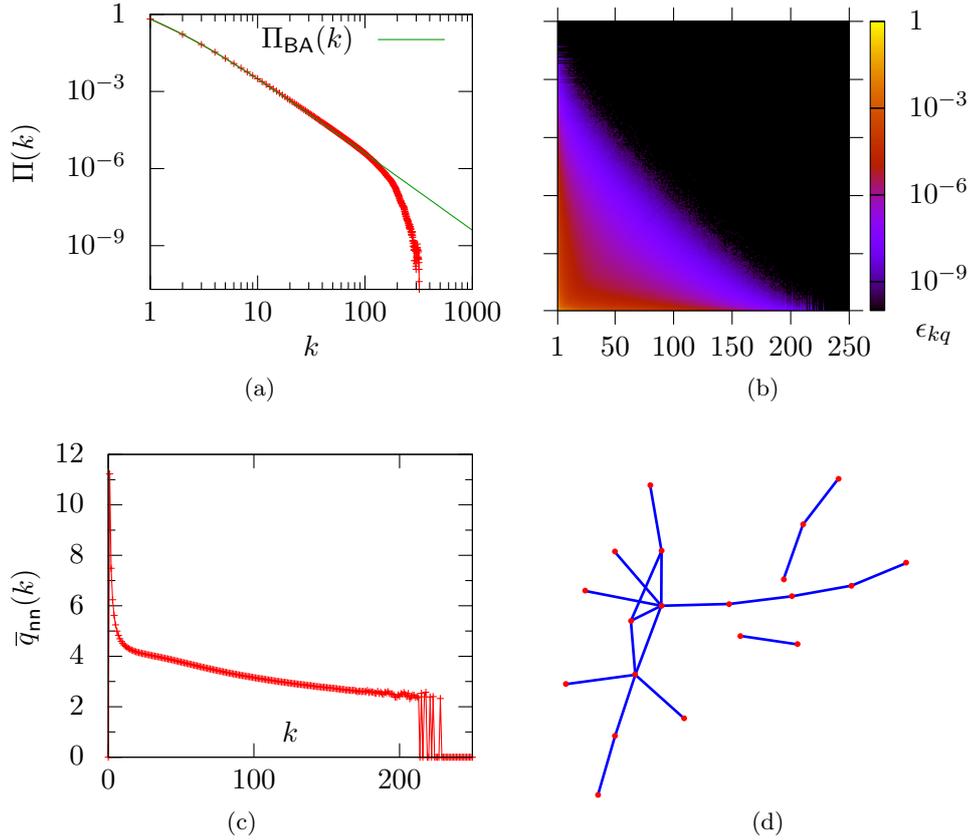


Figure 4.8.: Ensemble of scale-free equilibrated networks featuring the degree distribution known from Barabási–Albert graphs. To further approximate the Barabási–Albert model of growing networks the sampling might be restricted to tree graphs. The local node weight function used to define the ensemble is given in (4.15). The following quantities are obtained from a sample of $\mathcal{N} = 50 \times 10^6$ simple graphs with $L = N = 1000$ nodes and sites respectively: (a) the degree distribution in comparison to the exact distribution of the Barabási–Albert model, (b) degree-degree correlations, (c) average neighbor degree and (d) a sample with $N = 20$ and $L = N - 1 = 19$.

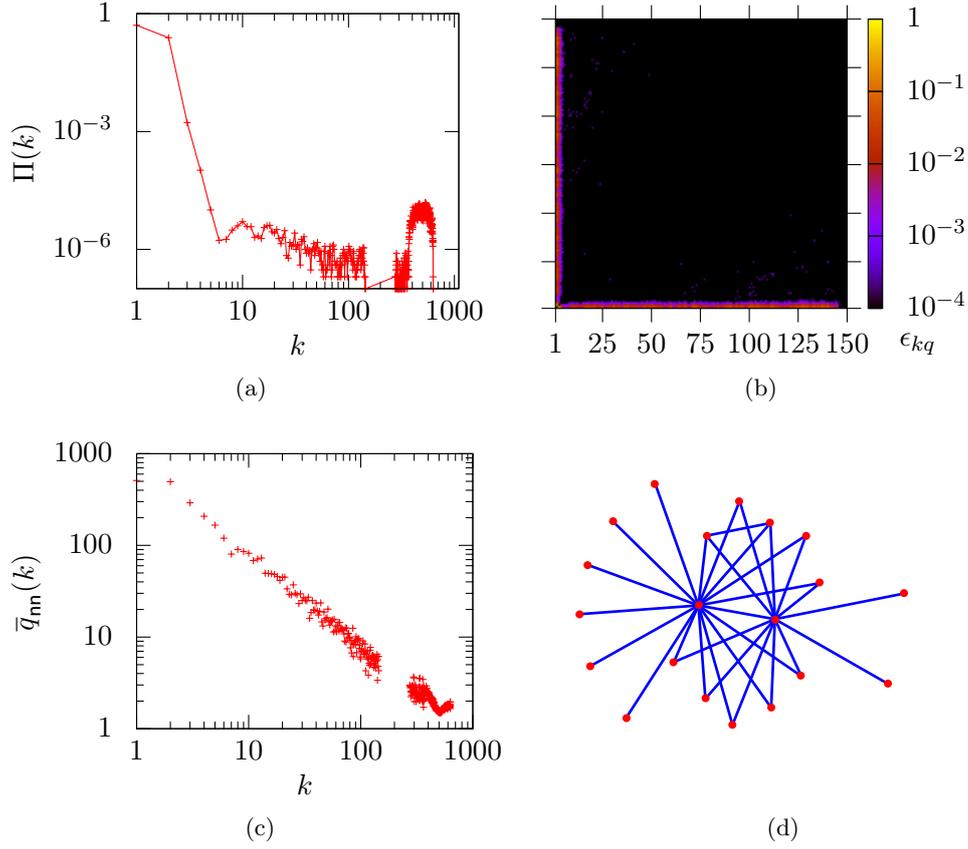


Figure 4.9.: Example data of an ensemble defined with the local edge weight function (4.36). The falloff at in the degree distribution (a) for low degrees ($q > 2$) and the peak at $q \sim 500$ suggest that star-like graphs are typical in this ensemble. The degree-degree correlations (b) and average neighbor degree (c) generally confirm this picture by indicating strong disassortativity. (d) shows a sample graph with $N = 20, L=30$ nodes and sites respectively. The properties were estimated from a sample of $\mathcal{N} = 50 \times 10^6$ graphs with $L = N = 1000$.

4. Monte Carlo Generation of Complex Networks

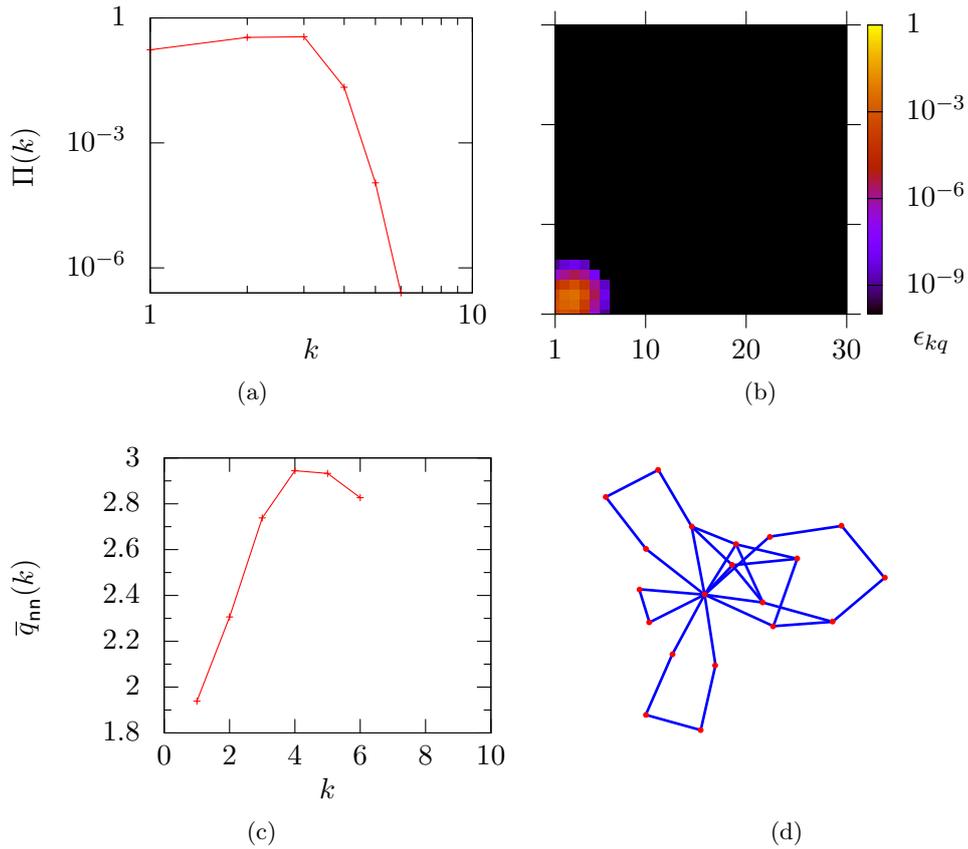


Figure 4.10.: Properties and a sample graph from the graph ensemble defined by the weights (4.37). The figures show (a) the degree distribution, (b) the degree-degree correlation function, (c) the average neighbor degree and (d) a sample graph ($L = 30, N = 20$). The desired feature of assortative graphs is confirmed in the plots of both ϵ_{kq} and q_{nn} . For the estimation of these properties $\mathcal{N} = 10000$ graphs with $L = N = 1000$ were generated.

5. Conclusion and Outlook

The condensation process that is observed in the PFSS process as a relaxation of the system into its steady state has been studied mainly using numerical techniques, to understand the underlying condensation mechanism and find its typical time scale.

It has been found that the condensation mechanism relies on the fast evaporation of small droplets in a first regime, leading to a coarse grained system with few condensates of similar mass. The further condensation is dominated by slightly biased fluctuations of the condensate masses leading to a slow transfer of particles to the largest condensate until it remains as the single condensate in the system. The expected time scale of this process is $\tau \propto M^\delta$ with a scaling exponent $\delta = 2$ for the asymmetric process and $\delta = 3$ for the symmetric process. These scaling exponents have been confirmed by means of numerical simulations of the condensation process. The estimated scaling exponents are $\delta = 2.04 \pm 0.07$ and $\delta = 2.9 \pm 0.2$ respectively. Hence, the time scale of the condensation in the PFSS process is the same as in the zero-range process.

The scaling of the condensation time for the symmetric transport process on lattices in two dimensions has been studied. It fulfills the same scaling law $\tau \propto M^\delta$ with the scaling exponent $\delta = 1.95 \pm 0.1$ that has been estimated by numerical computations. This agrees with the expected with the expected scaling behavior and exponent for the proposed mechanism of mass condensation and the condensation time scale of the zero-range process.

Due to the slow dynamics of the studied process and the requirement to directly simulate it, I was only able to systematically study the condensation process in relatively small systems of up to $M = N = 500$ sites and particles respectively in the totally asymmetric ($p = 1$) case of the model. As the dynamics of the symmetric process are much slower, exhibiting a scaling exponent of approximately $\delta \sim 3$, numerical investigation is limited to very small systems of about $N = 200$ in reasonable time. However, in systems of this size still strong finite size effects are present and affect observations.

Mass condensation in the the discussed transport process does indeed occur on lattices and regular graphs in two dimensions ($k = 4$ for the latter) with critical densities of $\rho_c = 0.00317 \pm 0.00005$ and $\rho_c \sim 0.14 \pm 0.02$ respectively. The dynamics on lattices lead to an extended condensate which has a similar shape as in the one-dimensional case. The scaling exponent is estimated as $\delta = 1.95 \pm 0.1$ which again is the same as for the ZRP. For regular graphs, it remains unclear whether the condensate is extended in the steady state. The observed average condensate size is in the order of 4 sites but might still decrease in much longer simulations.

For the process on regular graphs no scaling of the condensation time has been observed in this study. However, the condensation process seems to be

5. *Conclusion and Outlook*

dominated by the local structure of the graph, as the growth of the average mass of the biggest condensate exhibits plateaus with life-time of the order of the condensation time. As an explanation, the slow movement of the condensate to the region of the graph where the condensate is most stable is proposed. That is, to a cluster of nodes in the graph with few connections to the remaining network.

Another interesting feature of the discussed transport process that is worth further study is the movement of the condensates in the coarsening regime of the condensation process in one dimension. It has been observed in the site occupation time series that large condensates move “synchronized”, which indicates the existence of a long range interaction between them affecting their motion.

As a spin-off the graphgen package for Monte Carlo generation of equilibrated graphs was implemented and tested. It allows the explicit introduction of degree-degree correlations between the nodes of graphs, allowing for a finer tuning of the structural properties of the graphs to be generated. Specifically this may be used to enable further systematical study of the condensation process on graphs with defined structure properties. Furthermore, as the toolkit is designed generically, it is suitable for the general study of complex-network properties and the simulation of real-world networks.

A. Partition Function of the ZRP on scale-free Networks

The original partition function for the balls in boxes model can be rewritten as a recursion

$$\begin{aligned}
 Z(N, M) &= \sum_{m_1=0}^M \cdots \sum_{m_N=0}^M \delta_{M, \sum_{i=1}^N m_i} \prod_{i=1}^N p(m_i) q_i^{m_i} & (A.1) \\
 &= \sum_{m_N=0}^M p(m_N) q_N^{m_N} \left[\sum_{m_1=0}^{M-m_N} \cdots \sum_{m_{N-1}=0}^{M-m_N} \delta_{M, \sum_{i=1}^{N-1} m_i} \prod_{i=1}^{N-1} p(m_i) q_i^{m_i} \right] \\
 &= \sum_{m_N=0}^M p(m_N) q_N^{m_N} Z(N-1, M-m_N), & (A.2)
 \end{aligned}$$

with the the node degree q_i and the weight

$$p(m) = \prod_{n=1}^m \frac{1}{u(n)}, \quad p(0) = 1. \quad (A.3)$$

The hopping rate function

$$u(m) = \begin{cases} 0 & m = 0 \\ 1 & m \geq 1 \end{cases} \quad (A.4)$$

used in the study of condensate life-times in [22] simplifies the derivation of the recursion. The weights are then $p(m) = 1$ for all occupation numbers m , the factor disappears. Now the network structure information given as an unordered node degree sequence $\{q_i\}$ is rewritten to a degree multiplicity sequence $\{\lambda_q\}$ with $\lambda_q = \sum_i^N \delta_{q, q_i}$. The multiplicity sequence is proportional to the degree distribution, which is a power law for scale free networks. This is exploited by the recursive calculation of the partition function starting with the highest degrees in each step. Thus the recursion tree grows only at the order of the number of different degrees instead of the number of nodes as nodes of the same degree are grouped. The combinatoric multiplicity of these groups is the number of partitions of m identical particles on λ_q equal sites giving a binomial factor.

The resulting recursive partition function is

$$Z_k(M, \{\lambda_q\}) = \sum_{m=0}^M \binom{\lambda_k + m - 1}{m} k^m Z_{k-1}(M-m, \{\lambda'_k\}), \quad (A.5)$$

with the highest degree k of the remaining network and the truncated degree multiplicity sequence $\{\lambda'_k\}$, with $\lambda_{q>k} = 0$.

A. Partition Function of the ZRP on scale-free Networks

To finally compute the partition function, conditions to break the recursion are calculated for residue networks as follows:

$$\begin{aligned}
 Z(N = 1, M, \lambda_k) &= \sum_{m=0}^M \delta_{m, M} \prod_{i=1}^1 k_i^{m_i} p(m) \\
 &= k^M p(M) \\
 Z(N = \lambda_k, M) &= \binom{\lambda_k + m - 1}{m} k^M p(M)
 \end{aligned} \tag{A.6}$$

$$\begin{aligned}
 Z(N, M = 1, \{\lambda_k\}) &= \sum_{m_1=0}^M \cdots \sum_{m_N=0}^M \delta_{M, \sum_{i=1}^N m_i} \prod_{i=1}^N p(m_i) k_i^{m_i} \\
 &= \sum_{k=1}^{\infty} k \lambda_k p(1)
 \end{aligned} \tag{A.7}$$

$$\begin{aligned}
 Z(N, M = 0, \{\lambda_k\}) &= \sum_{m_1=0}^M \cdots \sum_{m_N=0}^M \delta_{M, \sum_{i=1}^N m_i} \prod_{i=1}^N p(m_i) k_i^{m_i} \\
 &= 1.
 \end{aligned} \tag{A.8}$$

In principle this works for any given degree sequence $\{q_i\}$, with the same constraints that (A.1) has, like the graph must be connected. For the computation of expectations using the recursion, the order of the multiplicity sequence does not matter in principle. However, ordering the sequence and entering the recursion with the highest remaining degree heavily reduces the calculation time. As a bonus, intermediate $Z(M - m, N - n)$ may be stored and reused over and over during the recursion.

B. Application of the Graphgen Library

In this appendix a minimal example program using the graphgen library is presented in Figure B.1. The goal is to demonstrate the practical interaction of modules supplemented as class templates to contrast the schematic view given in Figure B.2.

B. Application of the Graphgen Library

```
#include<ofstream>
#include"graphgen.hpp"

double f( qint q, qint k ){ return pow(q-k, 2); }

using namespace graphgen;
typedef undirected_graph Graph;
typedef link_weight<graphgen::value_type::weight> Weight;
typedef canonical<Weight, shape::simple> Ensemble;

const int NV=1000;

int main(){
    // create graph object and construct BA graph
    Graph graph(NV);
    graph.gen_BarabasiAlbert(1);
    // create ensemble object and estimator
    Weight weight(f);
    Ensemble E( graph, weight );
    degreedist Deg(2*NV);

    for( int n=0; n<1000000; n++ ){
        E.sweep();
        Deg(graph);
    }
    // write degree distribution and last graph to files
    std::ofstream degdstr("degree");
    degdstr<<Deg;
    std::ofstream gstream("graphs");
    gstream<<graphgen::graphviz(graph)<<std::endl;
}
```

Figure B.1.: Minimal example simulation program employing the graphgen library. The ensemble sampling class template is constructed at the beginning as a `typedef` and is then created as an object using the created weight bound to the node weight (4.36), and a graph object that is initialized as a BA graph (constructed with 1 link added per node). As an example, a degree distribution estimator is used. MC sweeps and measurements are done and the obtained degree distribution is written to a file. The last obtained graph configuration is written in GraphViz format.

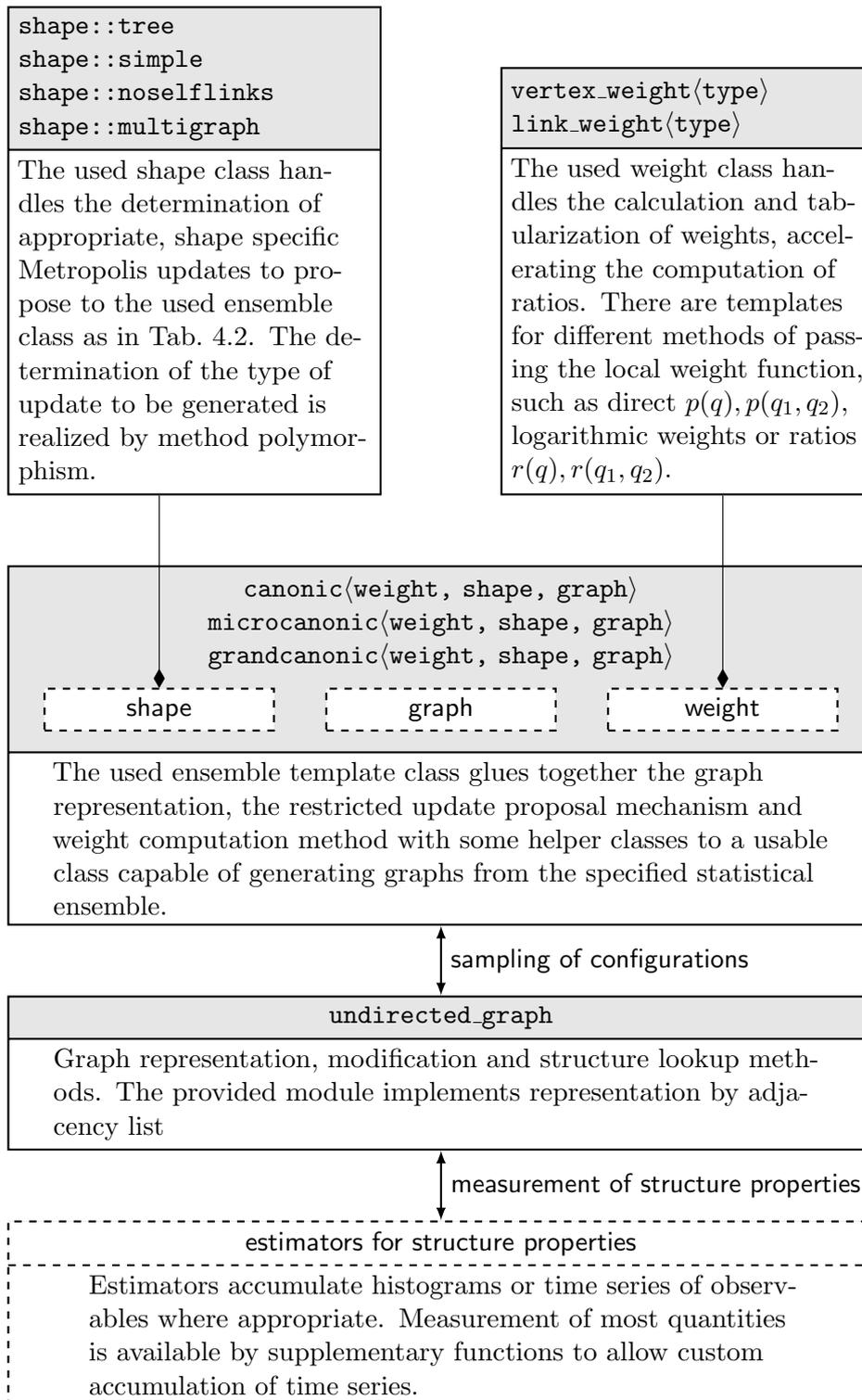


Figure B.2.: Scheme of interacting modules using combined template classes employed by the graphgen library. For a more practical understanding, consider to look up the technical documentation or example programs.

Bibliography

- [1] C. Narteau, E. Lajeunesse, F. Métivier, and O. Rozier, in *Modelling of dune patterns by short range interactions*, edited by Parker and García (2006).
- [2] J. Krug and T. Michely, *Islands, Mounds and Atoms: Patterns and Processes in Crystal Growth Far From Equilibrium* (2004).
- [3] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, *Critical phenomena in complex networks*, Rev. Mod. Phys. **80**, 1275 (2008).
- [4] W. Nolting, *Statistische Physik* (2007).
- [5] F. Schwabl, *Statistische Mechanik* (2006).
- [6] M. R. Evans, *Phase transitions in one-dimensional nonequilibrium systems*, Braz. J. Phys. **30**, 42 (2000).
- [7] D. P. Landau and K. Binder, *Monte Carlo Simulations in Statistical Physics* (2000).
- [8] N. Metropolis and S. Ulam, *The Monte Carlo Method*, J. Am. Stat. Assoc. **44**, 335 (1949).
- [9] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *Equation of State Calculations by Fast Computing Machines*, J. Chem. Phys. **21**, 1087 (1953).
- [10] W. Janke, in *Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms, NIC Series, Vol. 10*, edited by J. Grotendorst, D. Marx, and A. Muramatsu (2002), pp. 423–445, proceedings of the Euro Winter School.
- [11] J. Hoshen and R. Kopelman, *Percolation and cluster distribution. I. Cluster multiple labeling technique and critical concentration algorithm*, Phys. Rev. B **14**, 3438 (1976).
- [12] F. Babalievski, *Cluster counting: The Hoshen-Kopelman algorithm vs. spanning tree approaches*, Int. J. Mod. Phys. C **9**, 43 (1997).
- [13] C. Joas, Introduction to the Hoshen–Kopelman algorithm and its application to nodal domain statistics, 2006, union-find implementation of HK and introduction.
- [14] D. Chowdhury, L. Santen, and A. Schadschneider, *Statistical Physics of Vehicular Traffic and Some Related Systems*, Phys. Rep. A **329**, 199 (2000).

Bibliography

- [15] F. Spitzer, *Interaction of Markov processes*, Adv. Math **5**, 246 (1970).
- [16] P. Białas, Z. Burda, and D. A. Johnston, *Balls in Boxes and Quantum Gravity*, Nuc. Phys. B **63**, 763 (1998).
- [17] B. Waclaw, L. Bogacz, Z. Burda, and W. Janke, *Condensation in zero-range processes on inhomogeneous networks*, Phys. Rev. E **76**, 046114 (2007).
- [18] C. Godrèchet, *Dynamics of condensation in zero-range processes*, J. Phys. A **35**, 6313 (2003).
- [19] M. R. Evans and T. Hanney, *Nonequilibrium statistical mechanics of the zero-range process and related models*, J. Phys. A **38**, R195 (2005).
- [20] C. Godrèchet and J. M. Luck, *Dynamics of the condensate in zero-range processes*, J. Phys. A **38**, 7215 (2005).
- [21] L. Bogacz, Z. Burda, W. Janke, and B. Waclaw, *Balls-in-boxes condensation on networks*, Chaos **17**, 026112 (2007).
- [22] H. Nagel, *Balls in Boxes Condensation on scale-free Networks*, Theoretikum, 2007.
- [23] M. R. Evans, T. Hanney, and S. N. Majumdar, *Interaction driven real-space condensation*, Phys. Rev. Lett. **97**, 010602 (2006).
- [24] B. Waclaw, J. Sopik, W. Janke, and H. Meyer-Ortmanns, *Pair-factorized steady states on arbitrary graphs*, J. Phys. A **42**, 315003 (2009).
- [25] B. Waclaw, J. Sopik, W. Janke, and H. Meyer-Ortmanns, *Tuning the shape of the condensate in spontaneous symmetry breaking*, Phys. Rev. Lett. **103**, 080602 (2009).
- [26] B. Waclaw, J. Sopik, W. Janke, and H. Meyer-Ortmanns, *Mass condensation in one dimension with pair-factorized steady states*, J. Stat. Mech. **10**, P10021 (2009).
- [27] M. E. J. Newman, *Clustering and preferential attachment in growing networks*, Phys. Rev. E **64**, 025102 (2001).
- [28] L. A. N. Amaral, A. Scala, M. Barthélémy, and H. E. Stanley, *Classes of Small-World Networks*, Proc. Nat. Acad. Sci. USA **97** **97**, 11149 (2000), networks of conformations of proteins connected if they can be transformed by basic moves.
- [29] P. Erdős and A. Rényi, *On random graphs. I.*, Publ. Math. Debrecen **6**, 290 (1959).
- [30] B. Bollobás, *Random Graphs*, No. 73 in *Cambridge studies in advanced mathematics* (2001).
- [31] S. Milgram, *The small-world problem*, Psychology Today **2**, 60 (1967).

- [32] S. N. Dorogovtsev and J. F. F. Mendes, *Evolution of Networks* (2003).
- [33] R. Albert and A.-L. Barabási, *Statistical mechanics of complex networks*, Rev. Mod. Phys. **74**, 47 (2002).
- [34] B. Waclaw, *Statistical mechanics of complex networks*, 2007.
- [35] P. Hines, S. Blumsack, E. C. Sanchez, and C. Barrows, in *Proceedings of the 43rd Hawaii International Conference on System Sciences* (2010).
- [36] D. J. Watts and S. H. Strogatz, *Collective dynamics of 'small-world' networks*, Nature **393**, 440 (1998).
- [37] L. Bogacz, Z. Burda, W. Janke, and B. Waclaw, *A program generating homogeneous random graphs with given weights*, Comp. Phys. Comm. **173**, 162 (2005).
- [38] L. Bogacz, Z. Burda, and B. Waclaw, *Homogeneous complex networks*, Physica A **366**, 587 (2006).
- [39] B. Waclaw, L. Bogacz, and W. Janke, *Approaching the thermodynamic limit in equilibrated scale-free networks*, Phys. Rev. E **78**, 061125 (2008).

Danksagung

Zunächst möchte ich mich für die Betreuung während meiner Diplomarbeit, Diskussionen und Hilfestellung bei Prof. Wolfgang Janke und Dr. Bartłomiej Waclaw bedanken. Sie ermöglichten mir das Erarbeiten meiner Diplomarbeit in einer freundlichen und aufgeschlossenen Abteilung und nahmen sich Zeit für Fragen und Erörterungen.

Für Diskussionen und Impulse, die mir halfen den Kurs zu halten und zu verbessern danke ich meinen Kommilitonen Konrad Zimmermann, Michael Schellenberger Costa, Benjamin Werner, Sebastian Schöbl, Sebastian Sturm sowie Johannes Zierenberg.

Für ihre Hilfe bei der Suche nach Rechtschreibfehlern und einige Hinweise danke ich Dan Strehle und Michael Schellenberger Costa.

Meinen Eltern möchte ich für ihr Vertrauen und ihre Unterstützung meines Studiums das ich mit dieser Arbeit vorerst abschließen danke.

Großer Dank gebührt letztlich Lydia Gräfe, die mir insbesondere in der letzten Phase dieser Arbeit viel Zeit und Rückhalt gegeben hat.

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die Diplomarbeit selbständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Alle Stellen der Arbeit, die wörtlich oder sinngemäß aus Veröffentlichungen oder aus anderweitigen fremden Äußerungen entnommen wurden, sind als solche kenntlich gemacht. Ferner erkläre ich, dass die Arbeit noch nicht in einem anderen Studiengang als Prüfungsleistung verwendet wurde.

Hannes Nagel

Ich bin einverstanden, dass die Arbeit nach positiver Begutachtung in der Universitätsbibliothek zur Verfügung steht.

Hannes Nagel