Stochastic Modeling of Interacting Agent Systems

Luzie Helfmann

Master's Thesis

Fachbereich Mathematik und Informatik
Freie Universität Berlin

Submitted in December 2018
Revised in January 2019

Berlin Mathematical School

Freie Universität Berlin

Supervisor: Prof. Dr. Christof Schütte
Second Supervisor: Dr. Nataša Djurdjevac Conrad

Acknowledgments

First of all, I would like to thank my supervisor Christof Schütte for giving me the chance and supporting me to do research in this fascinating and interdisciplinary corner of mathematics. I would like to thank Nataša Djurdjevac Conrad and Stefanie Winkelmann for their research guidance and for always being there for me. Further, I would like to say my thanks to Ana Djurdjevac for answering all my questions on SPDEs, and to Changho Kim for an insightful conversation on the field of reaction-diffusion systems.
The CMD group at ZIB provided a great working atmosphere with engaging discussions making work and research much more enjoyable. Especially I want to say thanks to Alexander Sikorski, Johannes von Lindheim and Niklas Wulkow for proof reading my thesis and their valuable feedback.

Contents

List of Symbols and Abbreviations

We list the most important notation and abbreviations, the remaining notation will be introduced as we go.

$\mathbb{S}$	State space of a stochastic process
$\mathbb{T}$	Index set (often time) of a stochastic process
$(\Omega, \mathcal{F}, \mathbb{P})$	Probability space
$\mathcal{P}(t)$	Unit-rate Poisson process
$\mathbb{N}_0$	Natural numbers including 0
$\mathbb{E}(X)$	Expectation of a random variable X
$\mathrm{Var}(X)$	Variance of a random variable X
$\mathcal{U}(a, b)$	Uniform distribution on $[a, b]$
$B(t)$	Standard Brownian motion taking values in the real numbers
$\mathcal{N}(m, \sigma^2)$	Normal distribution with mean m and variance σ^2
$\mathrm{Exp}(\lambda)$	Exponential distribution with parameter λ
$Z(t)$	White noise process (either space-time or just in time)
$V(x)$	(Suitability) landscape, also called potential
∇	Gradient operator
Δ	Laplace operator
$W(t)$	Q- or cylindrical Wiener process
Q	Covariance operator of the Q-Wiener process
$L^2(D)$	Set of all functions $f : D \to \mathbb{R}$ such that $\left(\int_D \lvert f(x)\rvert^2 dx\right)^{\frac{1}{2}} < \infty$
M	Truncation for the expansion of the cylindrical Wiener process
$\mathrm{dom}(L)$	Domain of an operator L
$\mathcal{P}$	Orthogonal projection operator
N	Total number of agents
D	Domain of interest, subset of $\mathbb{R}^n$
N_T	Number of different agent types (species)
$X(t) = (X_i(t))_{i=1}^N$	Vector of positions for N agents at time t
$Y(t) = (Y_i(t))_{i=1}^N$	Vector of agent types (e.g. innovation state) at time t
N_R	Number of interaction rules for agents
$\{R_r\}$	Set of interaction rules, $r = 1, \ldots, N_R$
$U_i(x)$	Attraction-repulsion potential felt by agent i
$A(t)$	Adjacency matrix for the contact network at time t
d_{int}	Radius between agents at which they start to interact
T_s	Agent type (e.g. 'infected agent') for $s \in \{0, \ldots, , N_T\}$
$\lambda_i^r(t)$	Transition rate function for the ABM
$\gamma^r_{\mathrm{micro}}$ resp. γ^r_{meso}	Constant influence rate for rule R_r on the micro-scale resp. meso-scale
v_r	Type change due to the interaction rule R_r (ABM)
$\rho_s(x,t)$ or $\rho_s(t)$	Number density or number concentration of agents of type T_s
N_s	Number of agents of type T_s, such that $\sum_s N_s = N$
$\mathcal{D}$	(Stochastic) diffusion operator of the SPDE

$\mathcal{I}$	(Stochastic) interaction operator of the SPDE
δ_{ij}	Kronecker delta defined such that $\delta_{ij} = 1$ if $i = j$ and 0 else
$\delta(x - y)$	Dirac delta distribution (also sometimes function)
$a^r(\rho(t))$	Transition rate function for the density-based model
ν_s^r	Discrete number change of T_s agents due to rule R_r (density-based model)
$\rho(x,t)$ or $\rho(t)$	Number density or number concentration of agents
$C_0^\infty(D)$	Space of infinitely differentiable functions with compact support on D

ABM	Agent-based model
CLE	Chemical Langevin equation
FE	Finite element
SDE	Stochastic (ordinary) differential equation

1. Introduction

Modeling real-world dynamics is of great importance for the understanding, prediction and manipulation of the dynamical process of interest. Mathematically, one is therefore interested in the model analysis and inference of its parameters, as well as the accurate simulation and control of the model. With computers becoming more and more powerful, the simulation of detailed and complex model descriptions becomes possible. The vast amounts of data that are available in today's times can in turn be used to feed, evaluate and verify these models.

The modeling of realistic processes is nowadays often based on agent-based models (ABMs). These are model formulations in terms of discrete interacting entities, so-called agents (e.g. humans, companies, organizations). The paradigm shift to ABMs is primarily due to their flexible descriptions and the resulting easy incorporation of data.
An ABM typically consists of two components, the interacting and interdependent agents, and a surrounding environment that they can interact with [25, 35]. Agent-based models describe the system on the micro-scale, that is on the smallest scale, by imposing rules for each individual agent. These models can include a large number of different types of agents with complex behavioural traits. The hope is that the system shows emergent patterns, i.e. that the collective actions of many individual agents on the micro-scale, produce patterns on a larger scale.

In this thesis we are concerned with models describing systems of spatially distributed agents that move in space and interact whenever they are close-by. On the micro-scale, the mobility of agents is modeled as a random movement, whereby agents move with preferred direction to more suitable regions of their environment. Whenever agents are close to each other in space, they communicate and change their type (e.g. information state, health state, etc.) at a certain rate. Mathematically, the model description is formulated as a system of coupled diffusion processes and Markov jump processes for each agent [7]. Agent-based models of this class can for instance be found as models for infection spreading [5, 39], innovation spreading [7, 8], chemical reactions [12] and pattern formation [42].

We cannot solve these ABMs analytically, instead we simulate realizations of the modeled dynamics. For large and real-world dynamics, simulations become costly since they scale badly for increasing agent numbers. Further, the model formulation is stochastic, requiring many Monte Carlo simulations to make adequate predictions for observables of the system. Model reductions with a small approximation error are therefore necessary. Generally, agent systems can be described on different scales, from the smallest scale (the micro-scale) to the largest and coarsest scale (the macro-scale). When coarse-graining the model, that is aggregating model components, its complexity is reduced.

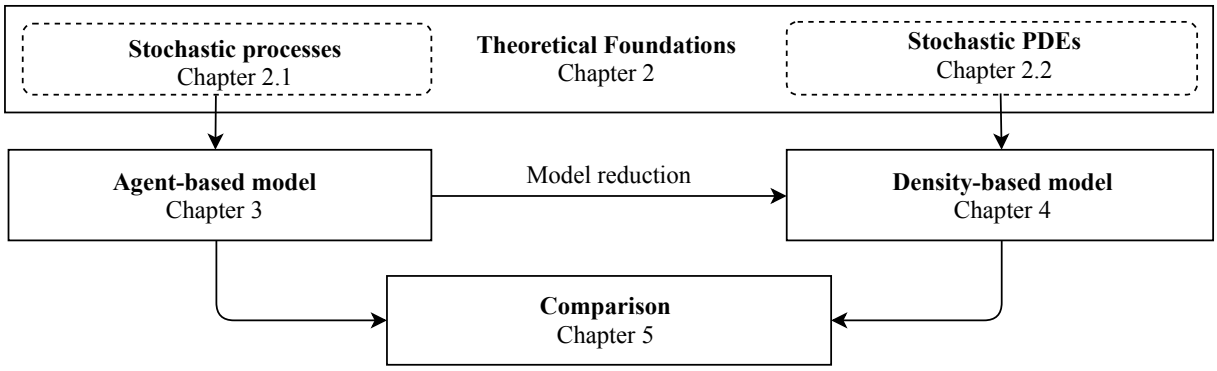
Here we present one model reduction technique for spatial systems of many agents and thus on the meso-scale [31, 10]. The idea is to replace agents of the same type by stochastic agent densities. These densities are interacting and diffusing in space according to a system of coupled stochastic PDEs (SPDEs). Classically, in the limit of infinitely many agents such dynamics are described by reaction-diffusion PDEs [41, 18]. But due to the finite number of agents, our sys-

tem is still inherently random. By adding fluctuations to the PDEs, the inherent stochasticity is captured by the model.
Deriving the reduced model, we closely follow the approaches in [31, 10], but we have added missing calculations and explanations and have extended the derivation to more complicated agent dynamics. Other approximation techniques exist for well-mixed (non-spatial) systems [19, 20] and for systems with discretized space [28, 48, 6]. For meso-scale systems of continuous space, the presented approach is the only one so far.

For the agent-based model as well as for the reduced model, we need numerical schemes for the efficient simulation of trajectories. For the time discretization of the ABM, we are proposing and explaining a coupled method [7] of Euler-Maruyama schemes for the diffusion processes and discretizations of the Markov jump processes (e.g. Temporal Gillespie algorithm [47] or discrete time methods).
Sampling trajectories of stochastic PDEs on the other hand, can be achieved by first discretizing in space and then discretizing in time using the Euler-Maruyama method. Space can be discretized on the basis of Finite Volume schemes [31, 13, 11] or by constructing a Finite Element discretization, as we are presenting in this thesis. The computational complexity of these methods is independent of the number of agents. The Finite Element approach is advantageous in our setting, since the stochastic PDEs have to be treated in the weak formulation framework due to a difficult noise term. Further many real-world systems contain complex boundaries that can be treated using irregular triangulations of the Finite Element method.

Last, we will study the ABM and the reduced SPDE model numerically on a toy example by comparing the computational effort and investigating the approximation quality of the reduced SPDE model to the ABM.

This thesis is structured as follows.



2. Theoretical Foundations

This chapter forms the theoretical basis for our agent-based model (Section 3) and the reduced model on the meso-scale (Section 4). The ABM is described by a system of coupled stochastic processes, a stochastic process describing the position dynamics and a process for the type changes of each agent. Therefore in the first half of this chapter we will introduce stochastic processes (taking values in the integers or real numbers) and their discretizations.
For systems of many agents, the ABM can be reduced to a system of stochastic PDEs. Each SPDE describes the evolution of the density for a certain agent type, i.e. the transport in space and the interactions with densities of other agent types. Thus the SPDEs are coupled. The solution of each SPDE can be considered as a stochastic process taking values in some infinite dimensional space. In the second half of this chapter we will introduce SPDEs and show how to numerically find solutions by first discretizing in space (Finite Element method) and then in time (Euler-Maruyama scheme).

2.1. Stochastic Processes and their Discretization

Stochastic processes are versatile and powerful descriptions forming the basis for many models of complex systems. The diffusion and reactions of molecular particles, the random fluctuations of a membrane, the behavior of the financial market or weather predictions, these are just some examples where modeling with stochastic processes is commonly used. A stochastic process can be viewed as a description for the evolution of a probabilistic system. Many real-world systems appear to be random due to inherent uncertainties.

In this section we will only introduce the definitions and facts about stochastic processes that are needed for describing our stochastic model in Chapter 3. Further, we give details of how to discretize and simulate trajectories of stochastic processes. For the basics of probability theory, we refer the reader to the literature [36]. A deeper treatment of stochastic processes and their discretizations can be found in [37, 40, 33].

Let us start by defining what a stochastic process is.

Definition 2.1.1. A *stochastic process* is a collection of random variables $X = \{X(t) : \Omega \to \mathbb{S}\}_{t\in\mathbb{T}}$ defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$[1].

The index set $\mathbb{T}$ is often called time and can be discrete, e.g $\mathbb{T} = \mathbb{N}$, $\mathbb{Z}$, or continuous, e.g. $\mathbb{T} = \mathbb{R}_+$, $[0, 1]$. The state space $\mathbb{S}$ can also be either discrete, e.g. $\mathbb{S} = \mathbb{N}$, or continuous such as $\mathbb{S} = \mathbb{R}^n$.

Another perspective on stochastic processes is to view them as a function of two variables, $X : \mathbb{T} \times \Omega \to \mathbb{S}$. For a fixed sample $\omega \in \Omega$, we call $X(\cdot, \omega)$ a sample path (realization, trajectory) of the process. In a computer experiment, one can view a fixed sample $\omega \in \Omega$ as the seed for

[1]The dependence of the stochastic process on ω, i.e. $\{X(t, \omega) : \Omega \to \mathbb{S}\}_{t\in\mathbb{T}}$, is often not explicitly written but should be clear from the context.

the random number generator of a simulation. For fixed $t \in \mathbb{T}$ on the other hand, $X(t, \cdot)$ is an $\mathbb{S}$-valued random variable.

In the following we will introduce two very important stochastic processes for modeling purposes, the Poisson process and Brownian motion.

2.1.1. The Poisson Process

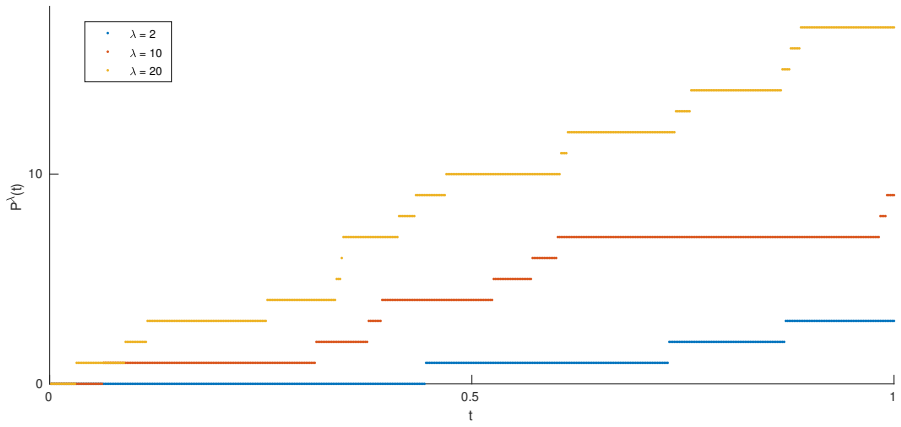


Figure 2.1.: Sample paths of the Poisson Process for different rates λ.

With the Poisson process we can model random and independent events in time, such as the arrival of people in a queue or the arrival of phone calls, by counting the number of random events that have happened up to some time point. As such the Poisson process is a continuous-time counting process $\{\mathcal{P}^\lambda(t) : \Omega \to \mathbb{N}_0\}_{t \geq 0}$. Counting processes have a discrete, non-negative state space $\mathbb{S}$ and are non-decreasing.

Definition 2.1.2. A *(time-homogeneous) Poisson process* with rate $\lambda > 0$ is a stochastic process $\{\mathcal{P}^\lambda(t) : \Omega \to \mathbb{N}_0\}_{t \geq 0}$ satisfying the following:

(i) $\mathcal{P}^\lambda(0) = 0$ (a.s.).

(ii) For $t > s \geq 0$, $\mathcal{P}^\lambda(t) - \mathcal{P}^\lambda(s)$ is Poisson distributed with parameter $\lambda(t-s)$, i.e.

$$\mathbb{P}(\mathcal{P}^\lambda(t) - \mathcal{P}^\lambda(s) = n) = \frac{(\lambda(t-s))^n}{n!} e^{-\lambda(t-s)}$$

for $n \in \mathbb{N}$.

(iii) The increments are independent.

See Figure 2.1 for some trajectories of the Poisson process with different rates (intensities) λ.

From the properties of the Poisson distribution, it immediately follows that

$$\mathbb{E}(\mathcal{P}^\lambda(t)) = \mathrm{Var}(\mathcal{P}^\lambda(t)) = \lambda t.$$

Another property of the Poisson process is that the waiting times τ, i.e. the time differences between events of the counting process, are exponentially distributed with mean $\frac{1}{\lambda}$. This can be deduced from the following.

The probability of having a waiting time τ until the next jump that is larger than s equals the probability of having no jump in a time interval of size s, i.e.

$$\mathbb{P}(\tau > s|\mathcal{P}^\lambda(t) = n) = \mathbb{P}(\mathcal{P}^\lambda(t+s) - \mathcal{P}^\lambda(t) = 0) = e^{-\lambda s}.$$

Hence the cumulative distribution function is $\mathbb{P}(\tau \leq s|\mathcal{P}^\lambda(t) = n) = 1 - e^{-\lambda s}$. The probability density of τ is the derivative (if it exists) of the cumulative distribution function, which in this case is just the density of the exponential distribution.

Denoting the unit-rate Poisson process (i.e. $\lambda = 1$) by $\mathcal{P}(t)$, it follows from Definition 2.1.2 that $\mathcal{P}^\lambda(t)$ is equivalent to a unit-rate process with scaled time, $\mathcal{P}(\lambda t)$.
So far we considered Poisson processes with constant rates $\lambda > 0$. By using the time-scaling argument, one can also define the so-called *inhomogeneous Poisson process* $\mathcal{P}(\int_0^t \lambda(s)ds)$ with time-dependent rate function $\lambda : [0,\infty) \to [0,\infty)$. For an inhomogeneous Poisson process the probability of $n \in \mathbb{N}$ jumps during the time interval $(t, t']$ is Poisson distributed with parameter $\int_t^{t'} \lambda(s)ds$. Further, $\int_t^{t+\tau} \lambda(s)ds$ is exponentially distributed[2] with rate 1 for two successive jump times t and $t+\tau$. This follows from a similar observation as for the homogeneous Poisson process.

Discrete-time Approximation

The goal is to find a time-discrete approximation $\{Y(t)\}_t$ to the homogeneous Poisson process $\{\mathcal{P}^\lambda(t)\}_{t\in[0,T]}$ forming the basis for the simulation of trajectories. First we discretize the time interval $[0,T]$ into small intervals of length Δt. Then instead of rates we consider probabilities per time interval.
From before we know that the probability of having at least one jump in a time interval of size Δt is just $1 - e^{-\lambda\Delta t}$ independently of the current value of the process. Further, it follows from Definition 2.1.2 (ii) that the probability of two or more jumps in a time interval Δt is $\mathcal{O}(\Delta t)$, using the little-o notation, which is thus negligible as Δt becomes very small. We assume the time interval Δt to be small enough such that it is a good approximation to have at most one jump per interval.
In pseudo-code the discrete time approximation reads as follows.

Initialize $Y(0) = 0$, $t = 0$. While $t < T$:

1. Draw $\theta \sim \mathcal{U}(0,1)$.

[2]The randomness enters $\int_t^{t+\tau} \lambda(s)ds$ via the integral limit since τ is a random variable.

2. If $\theta \leq 1 - e^{-\lambda \Delta t}$, the process jumps up by 1. Set $Y(t + \Delta t) = Y(t) + 1$.
   Else $Y(t + \Delta t) = Y(t)$.

3. Advance in time, $t = t + \Delta t$.

One can also simulate trajectories of the homogeneous Poisson process continuously in time, as we will explain next.

Continuous-time Scheme

We have seen before that the waiting times τ of $\{\mathcal{P}^{\lambda}(t)\}_{t\in[0,T]}$ are exponentially distributed, i.e. $\tau \sim \text{Exp}(\lambda)$. This property can be exploited to construct a statistically exact continuous-time simulation scheme. We can simulate continuous-time trajectories of the Poisson process by drawing i.i.d. waiting times τ_i, $i = 1, 2, \ldots$, and by letting the process have jumps at times τ_1, $\tau_1 + \tau_2$, $\tau_1 + \tau_2 + \tau_3$ etc.

Drawing random variables from an exponential distribution can be done by drawing random variables from a uniform distribution $\mathcal{U}(0, 1)$ and making use of the following theorem to transform them.

Theorem 2.1.3. (Inverse Transform Sampling [27]). Let the cumulative distribution function $F_X(x) = \mathbb{P}(X \leq x)$ of a continuous random variable X be strictly increasing. Then for $\theta \sim \mathcal{U}(0, 1)$, $F_X^{-1}(\theta)$ is a random variable with cumulative distribution function $F_X(x)$.

Proof. For a uniformly distributed random variable $\theta \sim \mathcal{U}(0, 1)$, we have $y = \mathbb{P}(\theta \leq y)$. Thus

$$\begin{aligned} F_X(x) &= \mathbb{P}(\theta \leq F_X(x)) \\ &= \mathbb{P}(F_X^{-1}(\theta) \leq x), \end{aligned}$$

where we used that F_X is invertible. It follows that the random variable $F_X^{-1}(\theta)$ has cumulative distribution function $F_X(x)$. □

In our case, we want to sample $\tau_i \sim \text{Exp}(\lambda)$ with the cumulative distribution function $F_{\tau_i}(s) = 1 - e^{-\lambda s}$ being strictly increasing on the positive real axis. Let us calculate the inverse,

$$\begin{aligned} \theta_i = F_{\tau_i}(s) &= 1 - e^{-\lambda s} \\ -\lambda s &= \log(1 - \theta_i) \\ s &= -\frac{1}{\lambda}\log(1 - \theta_i) = F_{\tau_i}^{-1}(\theta_i). \end{aligned}$$

Based on this we can simulate the waiting times τ_i between the $(i-1)^{\text{th}}$ and i^{th} jump. We generate $\theta_i \sim \mathcal{U}(0, 1)$ and set $\tau_i = -\frac{1}{\lambda}\log(1 - \theta_i)$. By Theorem 2.1.3, $\tau_i \sim \text{Exp}(\lambda)$ for all i.

Simulation of Inhomogeneous Poisson Processes

A Poisson process with a time-dependent rate function can be viewed as a unit-rate Poisson process with a scaled time axis $\{\mathcal{P}(\int_0^t \lambda(s)ds)\}_{t\in[0,T]}$. To sample realizations of the process, we

make use of the property $\int_t^{t+\tau} \lambda(s)ds \sim \text{Exp}(1)$ for the waiting times between two successive jump times t and $t+\tau$. By drawing samples from the exponential distribution with parameter 1, we can calculate the waiting times between jump events.
The pseudo-code for the continuous-time approximation $\{Y(t)\}_{t\in[0,T]}$ to the inhomogeneous Poisson process reads as follows.

Initialize $t = 0$, $Y(0) = 0$. While $t < T$:

1. Sample $\tilde{\tau} \sim \text{Exp}(1)$.
2. Numerically solve $\int_t^{t+\tau} \lambda(s)ds = \tilde{\tau}$ for τ (the waiting time between successive jumps).
3. Replace $Y(t+\tau) = Y(t) + 1$ and $t = t + \tau$.

2.1.2. Brownian Motion

Brownian motion was originally proposed by Robert Brown as a model for the erratic motion of a pollen grain immersed in water. Later the motion could be explained by the random collisions of the pollen grain with water molecules. The idea of Brownian motion can be put into a mathematical framework by using the concept of stochastic processes to describe the random position of the grain at time t and to model the motion as a continuous time stochastic process[3].

Definition 2.1.4. The *one-dimensional standard Brownian motion* $\{B(t) : \Omega \to \mathbb{R}\}_{t\geq 0}$ is a stochastic process with a.s. continuous paths such that

(i) $B(0) = 0$ (a.s.).

(ii) The increments are independent.

(iii) For every $t > s \geq 0$, the increment $B(t) - B(s) \sim \mathcal{N}(0, t-s)$.

In Figure 2.2 we show ten discretized trajectories of standard Brownian motion.

The standard n-dimensional Brownian motion $\{B(t) : \Omega \to \mathbb{R}^n\}_{t\geq 0}$ is the vector of n independent Brownian motions, i.e. $B(t) = (B_i(t))_{i=1,\dots,n}$ with $B_i(t)$ denoting independent one-dimensional Brownian motions. For the one-dimensional Brownian motion we have the expectation $\mathbb{E}(B(t)) = 0$ and variance $\text{Var}(B(t) - B(s)) = t - s$.
Further, a useful property of Brownian motion is the Markov property, that is the process is memoryless. Given the present, the future of the process does not depend on the past.

2.1.3. Stochastic (Ordinary) Differential Equations

As a next step, we are introducing (ordinary) differential equations driven by a noisy term that describe the evolution of a continuous-time stochastic process $\{X(t)\}_{t\in[0,T]}$.

[3]Often Brownian motion is also referred to as the Wiener process, here we stick to the term Brownian motion to avoid confusion with the Q-Wiener and cylindrical Wiener process that will be introduced later.

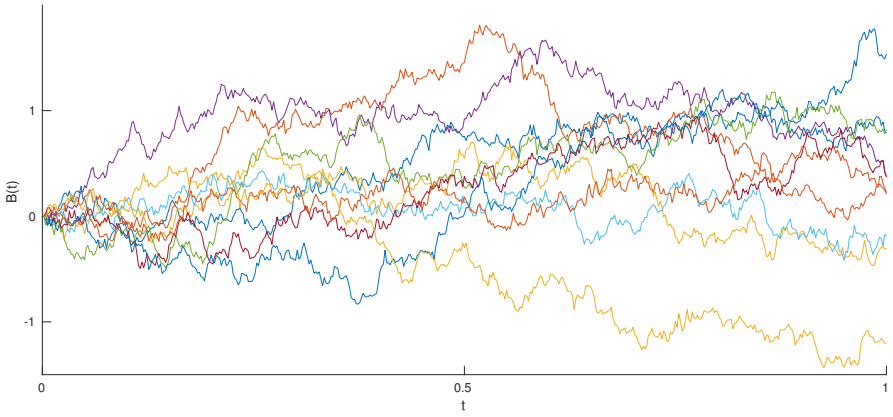


Figure 2.2.: Discretized sample paths of Brownian motion with $\Delta t = \frac{1}{1000}$.

For instance the stochastic differential equation (SDE)[4],

$$dX(t) = \mu dt + dB(t),\ X(0) = 0$$

describes Brownian motion with drift μ, i.e. $X(t) = \mu t + B(t)$.

Here we consider more general SDEs, so-called *(time-homogeneous) Itô diffusion processes*, of the form

$$\begin{aligned} dX(t) &= a(X(t))dt + b(X(t))dB(t) \\ X(0) &= X_0 \end{aligned} \tag{2.1.1}$$

with $X(t) \in \mathbb{R}^n$, $a : \mathbb{R}^n \to \mathbb{R}^n$ and $b : \mathbb{R}^n \to \mathbb{R}^{n\times d}$. $B(t)$ denotes the standard Brownian motion in $\mathbb{R}^d$. The first term is responsible for a deterministic drift, whereas the second term accounts for a noisy drift involving Brownian motion.

Since Brownian motion is nowhere differentiable [37, 40], the equation (2.1.1) is only symbolic and has to be interpreted as a stochastic integral equation

$$X(t) = X_0 + \int_0^t a(X(s))ds + \int_0^t b(X(s))dB(s),$$

where the last integral is the Itô integral [37, 40].

In the Physics literature one often writes the derivative of Brownian motion as the white noise process $Z(t) = \frac{dB(t)}{dt}$. The white noise process is a generalized stochastic process and uncorrelated at different time instances, such that $\mathbb{E}(Z_i(t)Z_j(s)) = \delta_{ij}\delta(t-s)$. Based on that, the SDE

[4] In this thesis when talking about SDEs we are only considering stochastic ordinary differential equation. Later we will introduce stochastic partial differential equations, in short SPDEs.

(2.1.1) can also be written as

$$\begin{aligned} \frac{dX}{dt}(t) &= a(X(t)) + b(X(t))Z(t) \\ X(0) &= X_0. \end{aligned} \tag{2.1.2}$$

Itô Formula

The chain rule for ordinary calculus becomes the Itô formula [37, 40] for Itô calculus by appending an additional term accounting for the uncertainty. We can use the Itô formula to find the differential of a function $f(t, X(t))$ depending on time t and the Itô process $\{X(t)\}_t$.

Theorem 2.1.5. (Itô Formula). For an Itô diffusion process $\{X(t)\}_t$ in one dimension and any $f \in C^2([0,T] \times \mathbb{R})$, one has that also $\{f(t, X(t))\}_t$ is an Itô diffusion process with[5]

$$df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial x}dX + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(dX)^2.$$

In order to compute the last term, the following rules have to be used

$$(dB)^2 = dt,\ dBdt = (dt)^2 = dtdB = 0.$$

For an n-dimensional Itô diffusion process $\{X(t)\}_t$ and any $f \in C^2([0,T] \times \mathbb{R}^n)$, the formula reads

$$df = \frac{\partial f}{\partial t}dt + \sum_{i=1}^{n}\frac{\partial f}{\partial x_i}dX_i + \frac{1}{2}\sum_{i,j=1}^{n}\frac{\partial^2 f}{\partial x_i \partial x_j}dX_i dX_j.$$

For a sketch of the proof we refer to [37].

Euler-Maruyama Discretization

We are seeking approximate numerical trajectories of SDEs of the form (2.1.1). The Euler-Maruyama method is similar to the explicit Euler scheme for ODEs, but also deals with the random term [32, 26].

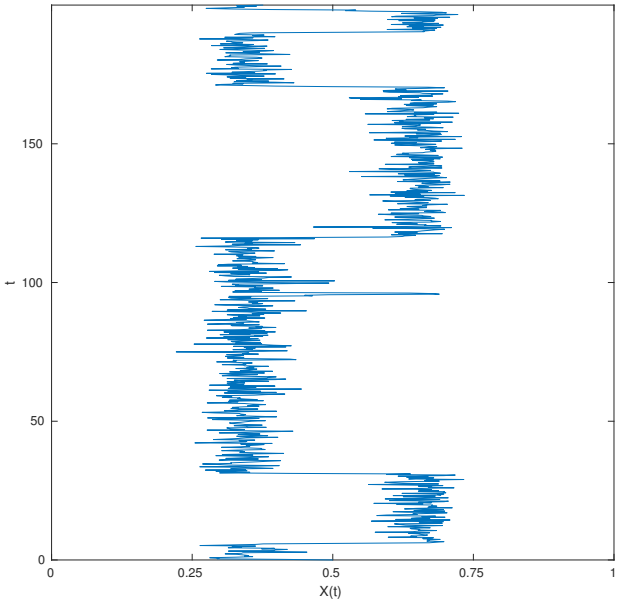
Discretizing time $[0,T]$ uniformly by $0 < \Delta t < 2\Delta t \cdots < T$ with $\Delta t = \frac{T}{K}$, we denote the Euler-Maruyama approximation by $\{Y(t)\}_{t\in\{0,\Delta t,\ldots,T\}}$. With the notation $Y_k = Y(k\Delta t)$, the approximation is iteratively defined for $k = 0, \ldots, K-1$ via

$$Y_{k+1} = Y_k + a(Y_k)\Delta t + b(Y_k)(B_{k+1} - B_k)$$

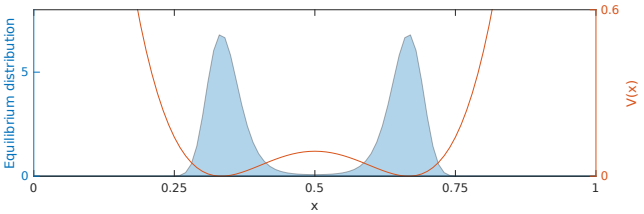
and with initial condition $Y_0 = X_0$.

The Brownian increments $\Delta B_k := (B_{k+1} - B_k)$ are independent and distributed normally. Therefore we can generate i.i.d. normal random variables $\Delta B_k \sim \mathcal{N}(0, \Delta t)$ for the Brownian increments.

[5] For ease of reading, we suppressed the dependency on $(t, X(t))$ and t sometimes.



(a) One realization of the diffusive motion in a double well potential starting at $X_0 = 0.5$ is plotted. The discretization of the SDE is based on the Euler-Maruyama method.



(b) Equilibrium distribution in the double well potential $V(x)$.

Figure 2.3.: The overdamped Langevin equation for a double well potential $V(x)$. The particle motion in the potential is metastable, the particle stays for a long time in the same well and only rarely transitions across the potential barrier.

Example: Overdamped Langevin Equation

The *overdamped Langevin equation* is an SDE modeling the random positional changes of particles (e.g. molecules [44], agents [7]) in a smooth potential $V(x)$, often called the energy landscape. More precisely, the particles are following Brownian motion with an additional drift given by a position-dependent force $-\nabla V(x)$. The dynamics are still memoryless. The particle

positions $\{X(t)\}_{t\in[0,T]}$ in $\mathbb{R}^n$ are described by the SDE

$$\begin{aligned} dX(t) &= -\nabla V(X(t))dt + \sigma dB(t) \\ X(0) &= X_0 \end{aligned}$$

with $B(t)$ denoting standard Brownian motion in $\mathbb{R}^n$, ∇ denotes the gradient operator, $V : \mathbb{R}^n \to \mathbb{R}$ gives the potential (also called landscape) and $\sigma \in \mathbb{R}$.

Let us have a closer look at an example system in one dimension. We are considering the random motion of a particle in the double well potential

$$V(x) = 0.1((6x-3)^2 - 1)^2.$$

The particle is randomly following the negative gradient of the potential. As such it is drawn towards the two minima of the landscape $V(x)$ at $x = \frac{1}{3}$ and $x = \frac{2}{3}$. One discretized trajectory of $\{X(t)\}_{t\in[0,200]}$ in the double well potential and with $\sigma = 0.2$ is plotted in Figure 2.3a. We can observe that the particle motion stays for a long time in the same well, with rare transitions between the two wells. The randomness in the motion enables these rare transitions.

The Fokker-Planck equation is a PDE describing the time evolution of the probability density function $p(x,t)$ of $X(t)$ [40]. For the overdamped Langevin equation, the Fokker-Planck equation is also called Smoluchowski equation and reads

$$\begin{aligned} \frac{\partial p}{\partial t}(x,t) &= \nabla \cdot (\nabla V(x) p(x,t)) + \frac{\sigma^2}{2}\Delta p(x,t) \\ p(x,0) &= p_0(x). \end{aligned} \tag{2.1.3}$$

For smooth confining[6] potentials, such as our double well potential, a unique equilibrium distribution exists [40] and is of the form

$$p_{\text{invariant}}(x) = \frac{1}{Z}\exp\left(-\frac{2V(x)}{\sigma^2}\right),$$

where Z is the normalization constant. For our double well example, the equilibrium distribution is concentrated around the two wells as depicted in Figure 2.3b.

One can also give an alternative interpretation of the Fokker-Planck equation. We can think of (2.1.3) as an equation describing the evolution of a density $p(x,t)$ of infinitely many non-interacting particles.

[6] A potential $V(x)$ is called confining if it goes to infinity for $|x| \to \infty$ and $\exp(-2V(x)/\sigma^2) \in L^1(\mathbb{R}^n)$ [40].

2.2. Discretizing Stochastic PDEs

We have seen what stochastic (ordinary) differential equations are. In this section we give an introduction and the necessary theory needed for treating stochastic PDEs (SPDEs). Since random fluctuations are inherent to natural phenomena and real-world processes, modeling with SPDEs is much more realistic than modeling real-world dynamics with deterministic PDEs. SPDEs can be used to describe e.g. the interface between different materials, the propagation of a signal along neurons or the flow of a fluid.

One of the most studied SPDEs is the stochastic heat equation describing the variation in temperature $u(x,t)$ in space and time. The stochastic heat equation in one dimension with a space-time white noise (STWN) forcing $Z(x,t)$ reads

$$\frac{du}{dt}(x,t) = \frac{d^2u}{dx^2}(x,t) + Z(x,t),\ x \in D \subseteq \mathbb{R},\ t \geq 0. \tag{2.2.1}$$

The space-time white noise process is uncorrelated (i.e. white) in space and time, such that $\mathbb{E}(Z(x,t)Z(y,t')) = \delta(x-y)\delta(t-t')$.

The stochastic heat equation can also be viewed as a stochastic ODE on some infinite dimensional function space. With $\{u(t)\}_{t\geq 0}$ denoting a stochastic process taking values in some function space on D, we can rewrite (2.2.1) as

$$\frac{du}{dt}(t) = \frac{d^2u}{dx^2}(t) + Z(t). \tag{2.2.2}$$

SPDEs modeling real-world dynamics are often driven by space-time white noise, i.e. by the stochastic process $\{Z(t)\}_{t\geq 0}$. For instance, the SPDE model approximating the agent-based dynamics for large population sizes (Section 4) contains two noise terms involving space-time white noise.

Similar as in the case of SDEs (Section 2.1.3), we want to study SPDEs as integral equations in time. We are therefore interested in defining a process $\{W(t)\}_{t\geq 0}$ that is Brownian in time and whose time derivative is space-time white noise, i.e. $Z(t) = \frac{dW}{dt}(t)$. With that we are able to rewrite the stochastic heat equation (2.2.2) as an integral equation

$$du(t) = \frac{d^2u}{dx^2}(t)\ dt + dW(t). \tag{2.2.3}$$

In Section 2.2.1 we will introduce the Q- and the cylindrical Wiener process taking values in a Hilbert space U. The time derivative of the cylindrical Wiener process will be space-time white noise. But the definition of the Q-Wiener process is needed to make sense of the cylindrical Wiener process. In Section 2.2.2 we will study semilinear evolution SPDEs driven by space-time white noise and explain how to discretize them.

2.2.1. The Q- and Cylindrical Wiener Process

For a separable Hilbert space U (i.e. U has a countable orthonormal basis), we define the U-valued Q-Wiener process and the U-valued cylindrical Wiener process and show how they can be expanded in some orthonormal basis [34]. They generalize the concept of real-valued Brownian motion to a stochastic process taking values in some infinite dimensional function space, in this case U.

The Q-Wiener process is correlated in space (i.e. coloured noise in space) with covariance operator Q. The cylindrical Wiener process on the other hand is uncorrelated in space (i.e. white in space). In time, both processes are Brownian with independent and normally distributed increments.

Definition 2.2.1. **[34]** Let $(\Omega, \mathcal{F}, \mathcal{F}_t, \mathbb{P})$ be a filtered probability space. Let the covariance operator $Q : U \to U$ be linear, bounded, non-negative definite and symmetric, such that Q has an orthonormal basis $\{\chi_m\}_{m\in\mathbb{N}}$ of eigenfunctions with eigenvalues $q_m \geq 0$ and $\sum_{m\in\mathbb{N}} q_m < \infty$. Then the U-valued stochastic process $\{W(t)\}_{t\geq 0}$ is a *Q-Wiener process* if

(i) $W(0) = 0$ (a.s.).

(ii) $W : \mathbb{R}^+ \to U$ is a continuous function for each $\omega \in \Omega$.

(iii) $W(t)$ is $\mathcal{F}_t$-adapted[7], and for $s < t$ the increment $W(t) - W(s)$ is independent of $\mathcal{F}_s$.

(iv) $W(t) - W(s) \sim N(0, (t-s)Q)$ for all $0 \leq s \leq t$[8].

The Q-Wiener process $\{W(t)\}_{t\geq 0}$ can be expanded in the orthonormal basis $\{\chi_m\}_{m\in\mathbb{N}}$ of Q with random coefficients. These expansions will be useful for simulating realizations of the stochastic process, but are also handy for proving theorems.

Theorem 2.2.2. *Let Q satisfy the assumptions from the Definition 2.2.1. Then $\{W(t)\}_{t\geq 0}$ is a Q-Wiener process if and only if*

$$W(t) = \sum_{m=1}^{\infty} \sqrt{q_m}\chi_m B_m(t), \text{ a.s.}$$

with $B_m(t)$ i.i.d. $\mathcal{F}_t$-Brownian motions and the series converges in $L^2(\Omega, U)$.

For a proof, we refer to [34].

Since we are ultimately interested in studying SPDEs driven by space-time white noise, we are looking for a U-valued process $\{W(t)\}_{t\geq 0}$ that is Brownian in time and whose time derivative is formally space-time white noise, i.e. $Z(t) = \frac{dW}{dt}(t)$. Space-time white noise is uncorrelated in space, such that the covariance operator $Q = I$ with eigenvalues $q_m = 1$ for all m. Hence

[7] That is, $W(t)$ is $\mathcal{F}_t$ measurable for each t.

[8] With a normal distribution N on a Hilbert space, i.e. $\mathbb{E}(W(t) - W(s)) = 0$ and the covariance operator of $W(t) - W(s)$ is $(t-s)Q$ [34].

$\sum_m q_m = \infty$ and $\{W(t)\}_{t\geq 0}$ is not a Q-Wiener process. What is more, the series expansion does not converge in $L^2(\Omega, U)$ anymore.
As a work-around, we will define the cylindrical Wiener process whose derivative in time is space-time white noise.

Definition 2.2.3. [34] The *cylindrical Wiener process* is the U-valued stochastic process $\{W(t)\}_{t\geq 0}$ defined by

$$W(t) = \sum_{m=1}^{\infty} \chi_m B_m(t) \text{ (a.s.)},$$

where $\{\chi_m\}_{m\in\mathbb{N}}$ is any orthonormal basis of U and $B_m(t)$ are mutually independent $\mathcal{F}_t$-adapted Brownian motions.

If $U \subset U_1$ for a larger Hilbert space U_1, then the series converges in $L^2(\Omega, U_1)$ as long as the inclusion $i : U \to U_1$ is a Hilbert-Schmidt operator [34]. When extending the process to the larger Hilbert space U_1, it can be shown to be a Q_1-Wiener process, where Q_1 is the covariance operator extended to U_1.

Making use of the cylindrical Wiener process expansion 2.2.3, we will in the following discretize sample paths of space-time white noise in one dimension and show that they are indeed uncorrelated in space and time.

Discretization of Sample Paths of Space-time White Noise on $[0, a]$

Space-time white noise can be sampled by numerically differentiating the truncated expansion of the cylindrical Wiener process in time [34]. As an orthonormal basis for the Hilbert space $L^2([0, a])$, we can take $\chi_m(x) = \sqrt{\frac{2}{a}}\sin\left(\frac{\pi m x}{a}\right)$, $m \in \mathbb{N}$. Then the truncated expansion to M terms $W^M(t)$ of the cylindrical Wiener process reads

$$W^M(t) = \sum_{m=1}^{M} \sqrt{\frac{2}{a}}\sin\left(\frac{\pi m x}{a}\right) B_m(t). \tag{2.2.4}$$

Discretizing in time with steps Δt, we get

$$W^M(t+\Delta t) - W^M(t) = \sum_{m=1}^{M} \sqrt{\frac{2}{a}}\sin\left(\frac{\pi m x}{a}\right) (B_m(t+\Delta t) - B_m(t))$$

with i.i.d. Brownian motions $B_m(t)$. The Brownian increments are normally distributed, i.e. $\zeta_m(t) := \frac{1}{\sqrt{\Delta t}} (B_m(t+\Delta t) - B_m(t)) \sim \mathcal{N}(0, 1)$, leading to

$$W^M(t+\Delta t) - W^M(t) = \sqrt{\Delta t} \sum_{m=1}^{M} \sqrt{\frac{2}{a}}\sin\left(\frac{\pi m x}{a}\right) \zeta_m(t).$$

Space-time white noise $Z(t) = \frac{dW}{dt}(t)$ is a mean zero process with covariance

$$\mathbb{E}\left(Z(x,t)Z(x',t')\right) = \delta(x-x')\delta(t-t'),$$

where δ denotes the Dirac Delta distribution. We are interested in computing the mean and covariance of the approximation

$$Z^M(t) = \frac{W^M(t+\Delta t) - W^M(t)}{\Delta t} = \sqrt{\frac{2}{a\Delta t}} \sum_{m=1}^{M} \sin\left(\frac{\pi m x}{a}\right) \zeta_m(t)$$

with the aim of checking whether the properties of $\{Z(t)\}$ carry over to $\{Z^M(t)\}$. To simplify computations, we are assuming $\Delta x M = a$. We denote spatial grid points by $x_i = i\Delta x$ and temporal grid points by $t_k = k\Delta t$. Following [34] we get for the covariance of $Z^M(t)$

$$\begin{aligned}
\mathbb{E}(Z^M(x_i,t_k)Z^M(x_j,t_l)) &= \frac{2}{a\Delta t}\mathbb{E}\left(\sum_{m=1}^{M} \sin\left(\frac{\pi m x_i}{a}\right)\zeta_{m,k} \sum_{m'=1}^{M} \sin\left(\frac{\pi m' x_j}{a}\right)\zeta_{m',l}\right) \\
&= \frac{2}{a\Delta t}\delta_{kl} \sum_{m=1}^{M} \sin\left(\frac{\pi m x_i}{a}\right)\sin\left(\frac{\pi m x_j}{a}\right) \\
&= \frac{1}{a\Delta t}\delta_{kl} \sum_{m=1}^{M} \left(\cos\left(\frac{\pi m(i-j)}{M}\right) - \cos\left(\frac{\pi m(i+j)}{M}\right)\right) \\
&= \frac{M}{a\Delta t}\delta_{kl}\delta_{ij} = \frac{1}{\Delta x \Delta t}\delta_{kl}\delta_{ij}, \qquad (2.2.5)
\end{aligned}$$

where δ denotes the Kronecker delta. In the first line we made use of

$$\mathbb{E}(\zeta_{m,k}\ \zeta_{m',l}) = \delta_{mm'}\delta_{kl},$$

in the second line we used the trigonometric addition theorems, and the last line follows from a property of the cosine function:

$$\sum_{m=1}^{M} \cos\left(\frac{\pi m q}{M}\right) = \begin{cases} M & q = 0 \\ 0 & q \text{ even and } q \neq 0 \\ -1 & q \text{ odd.} \end{cases}$$

The mean of the discretized STWN is zero, since the mean of $\zeta_m(t)$ is zero. From (2.2.5) we deduce that the discretized process $Z^M(t)$ is uncorrelated at different time instances and points in space. At the same point in space and time the process is correlated with an appropriate scaling to account for the grid sizes. We can conclude that the mean and covariance are consistent between the STWN process $\{Z(t)\}$ and its approximation $\{Z^M(t)\}$.

2.2.2. Semilinear Evolution SPDEs Driven by Space-time White Noise

In this section, we consider semilinear stochastic evolution PDEs that contain a random forcing in terms of space-time white noise $Z(t)$. Stochastic evolution PDEs describe the evolution of a random system in time and can be considered as stochastic ODEs on some infinite dimensional function space. We consider semilinear equations that are the sum of a linear and a nonlinear term. A deeper introduction into SPDEs can be found in [24, 9].

We are interested in computing path-wise realizations (i.e. for fixed samples $\omega \in \Omega$) of the solution ρ to a stochastic evolution PDE. The idea is to understand ρ as a stochastic process indexed by time and taking values in some infinite-dimensional function space. In our case this is a Hilbert space H containing functions on the domain D and with inner product $\langle \cdot, \cdot \rangle$. We consider the following evolution SPDE written as a stochastic ODE acting on $\rho(t)$[9]

$$\begin{aligned} \frac{d\rho}{dt}(t) &= -\mathcal{A}\rho(t) + F(\rho(t)) + G(\rho(t))Z(t) \\ \rho(0) &= \rho_0 \in H, \end{aligned} \tag{2.2.6}$$

with random forcing $Z(t) = \frac{dW}{dt}(t)$ for a U-valued cylindrical Wiener process $W(t)$. Thus $Z(t)$ is white (i.e. uncorrelated) in time and in space.

We suppose that the operator $-\mathcal{A} : \mathrm{dom}(\mathcal{A}) \subset H \to H$ is linear and has a complete orthonormal set of eigenfunctions and positive eigenvalues such that it generates a semigroup $S(t) = e^{-\mathcal{A}t}$. Then $-\mathcal{A}$ is called the infinitesimal generator. The boundary conditions (e.g. Dirichlet or Neumann boundary conditions) are incorporated into $-\mathcal{A}$. To give an example, the Laplacian $-\mathcal{A} = \Delta : \mathrm{dom}(\Delta) = H^2(D) \cap H_0^1(D) \subset L^2(D) \to L^2(D)$ fulfills these conditions and is therefore the generator of a semigroup [34].
The term $F : H \to H$ is non-linear, further $G : H \to \mathcal{HS}(U, H)$[10] can be non-linear. We assume that both F and G fulfill global Lipschitz conditions (as given in [34], Theorem 3.29 and Assumption 10.23).

There are different solution concepts for SPDEs. In all cases, we interpret the SPDE (2.2.6) as an integral equation in time (similar to SDEs). The necessary stochastic integral theory for Q- and cylindrical Wiener processes can be found in [34, 9].

Definition 2.2.4. [34] A predictable H-valued process $\{\rho(t)\}_{t\in[0,T]}$ is called a *strong solution* of (2.2.6) if

$$\rho(t) = \rho_0 + \int_0^t \left(-\mathcal{A}\rho(t') + F(\rho(t'))\right) dt' + \int_0^t G(\rho(t'))dW(t'), \ \ \forall t \in [0,T].$$

Since the function spaces for this formulation are very restrictive, i.e. $\rho(t) \in \mathrm{dom}(\mathcal{A})$ is needed, we will consider the weak solution framework and use it as a basis for the discretization of the SPDE.

Definition 2.2.5. [34] A predictable H-valued process $\{\rho(t)\}_{t\in[0,T]}$ is a *weak solution* of the SPDE (2.2.6) if for each $t \in [0,T]$

$$\begin{aligned} \langle \rho(t), w \rangle = &\langle \rho_0, w \rangle + \int_0^t \left(-\langle \rho(t'), \mathcal{A}w \rangle + \langle F(\rho(t')), w \rangle\right) dt' \\ &+ \int_0^t \langle G(\rho(t'))dW(t'), w \rangle, \ \ \forall w \in \mathrm{dom}(A), \end{aligned}$$

[9] The ω-dependence is not explicitly written here, but $\rho(t)$ is a H-valued stochastic process, i.e. $\{\rho(t) : \Omega \to H\}_t$.
[10] $\mathcal{HS}(U, H)$ is the set of Hilbert-Schmidt operators mapping from U to H [34].

where we consider the following expansion (based on Definition 2.2.3)

$$\int_0^t \langle G(\rho(t'))dW(t'), w\rangle = \sum_{m=1}^{\infty} \int_0^t \langle G(\rho(t'))\chi_m, w\rangle dB_m(t').$$

The requirements on $\rho(t)$ are now lifted to the test functions w, which needs much less regularity for $\rho(t)$ and is easier to work with. Strong solutions are usually also weak solutions, the reverse only holds under certain regularity conditions.

2.2.3. Discretization

Since we cannot solve the given evolution SPDE (2.2.6) analytically, we instead want to discretize the SPDE in order to sample trajectories of the discretized SPDE. We need to discretize in space and in time as well as dealing with the random term. In this section we are following the approach in [34].
Making use of the method of lines, we will first discretize in space and then in time. For the space discretization, the Galerkin method can be employed. Building on the weak solution, the SPDE is thereby approximated by a system of SDEs. The noise term can be expanded in some orthonormal basis. By truncating the basis to M terms, the noise term is projected onto a finite-dimensional subspace. We then discretize the system of SDEs in time using a semi-implicit Euler-Maruyama method.

Galerkin Approximation

With the assumptions on F, G, $-\mathcal{A}$ of the SPDE (2.2.6), it can be shown [34] that a weak formulation with solution and test function space $\mathrm{dom}(\mathcal{A}^{1/2})$ exists. This is for example the case for the stochastic heat equation (2.2.3) forced by space-time white noise. Based on the weak formulation of the SPDE, the Galerkin approximation consists of finding an approximate solution in some finite-dimensional subspace $\tilde{V} \subset \mathrm{dom}(\mathcal{A}^{1/2})$. For both, the solution and test functions, we use an $n+1$-dimensional subspace $\tilde{V}$ spanned by the basis $\{\phi_j\}_{j=0}^n$.

The Finite element (FE) method constructs the basis of $\tilde{V}$ by partitioning the domain into non-overlapping elements and defining a set of polynomial functions piecewise on the elements such that they are globally continuous. Further background theory on the Finite element method can be found in [29].

Let us now write down the Galerkin approximation for (2.2.6). We are searching $\tilde{\rho}(t) \in \tilde{V}$ such that

$$\begin{aligned}\langle \tilde{\rho}(t), \tilde{w}\rangle =& \langle \tilde{\rho}_0, \tilde{w}\rangle + \int_0^t \left(-a(\tilde{\rho}(t'), \tilde{w}) + \langle F(\tilde{\rho}(t')), \tilde{w}\rangle\right) dt' \\ &+ \int_0^t \langle G(\tilde{\rho}(t'))dW(t'), \tilde{w}\rangle, \quad \forall \tilde{w} \in \tilde{V},\ \forall t \in [0,T] \end{aligned} \tag{2.2.7}$$

with[11] inner product $a(u,v) := \langle \mathcal{A}^{1/2}u, \mathcal{A}^{1/2}v\rangle$ for $u,v \in \mathrm{dom}(\mathcal{A}^{1/2})$. The initial condition is given by $\tilde{\rho}_0 = \tilde{\mathcal{P}}\rho_0$, where $\tilde{\mathcal{P}} : H \to \tilde{V}$ denotes the orthogonal projection onto the finite-dimensional space $\tilde{V}$.
Since $\tilde{V}$ is spanned by $\{\phi_i\}_{i=0}^n$ by definition, requiring (2.2.7) is equivalent to

$$\begin{aligned}\langle \tilde{\rho}(t), \phi_i\rangle =& \langle \tilde{\rho}_0, \phi_i\rangle + \int_0^t \left(-a(\tilde{\rho}(t'), \phi_i) + \langle F(\tilde{\rho}(t')), \phi_i\rangle\right) dt' \\ &+ \int_0^t \langle G(\tilde{\rho}(t'))dW(t'), \phi_i\rangle, \quad \forall i = 0,\dots,n,\ \forall t \in [0,T]. \end{aligned} \tag{2.2.8}$$

Further with $\{\tilde{\rho}(t)\}_{t\in[0,T]}$ being a $\tilde{V}$-valued stochastic process, we can expand a realization of $\tilde{\rho}(t)$ as a linear combination

$$\tilde{\rho}(t) = \sum_{j=0}^n \beta_j(t)\phi_j$$

of the basis functions ϕ_j with time-dependent coefficients. Inserting into (2.2.8), we arrive at

$$\begin{aligned}\sum_{j=0}^n \beta_j(t)\langle \phi_j, \phi_i\rangle =& \sum_{j=0}^n \beta_j(0)\langle \phi_j, \phi_i\rangle + \int_0^t \left(-\sum_{j=0}^n \beta_j(t')a(\phi_j,\phi_i) + \langle F\left(\tilde{\rho}(t')\right), \phi_i\rangle\right) dt' \\ &+ \int_0^t \langle G\left(\tilde{\rho}(t')\right) dW(t'), \phi_i\rangle, \quad \forall i = 0,\dots,n,\ \forall t \in [0,T].\end{aligned}$$

Defining matrices $C_{ji} = \langle \phi_j, \phi_i\rangle$, $A_{ji} = a(\phi_j,\phi_i)$ for $i,j = 0,\dots,n$ and coefficient vector $\beta(t) = (\beta_j(t))_{j=0}^n$, we can write

$$\begin{aligned}\sum_{j=0}^n d\beta_j(t)C_{ji} =& \left(-\sum_{j=0}^n \beta_j(t)A_{ji} + \langle F\left(\tilde{\rho}(t)\right), \phi_i\rangle\right) dt \\ &+ \langle G\left(\tilde{\rho}(t)\right) dW(t), \phi_i\rangle, \quad \forall i = 0,\dots,n,\ \forall t \in [0,T].\end{aligned} \tag{2.2.9}$$

This equation is still understood as an integral equation.

We will in the following project the cylindrical Wiener process onto a finite-dimensional space in order to circumvent that $G(\tilde{\rho(t)})$ is acting on a stochastic process taking values in some infinite-dimensional space U, which is difficult to implement. By inserting an orthogonal projection operator $\mathcal{P}^M : U \to \mathrm{span}\{\chi_m\}_{m=1}^M$ in front of the U-valued Wiener process expansion, we project the process onto its first M basis functions and thus truncate the expansion to M terms:

$$\begin{aligned}\mathcal{P}^M W(t) = \mathcal{P}^M \sum_{m=1}^\infty \chi_m B_m(t) &= \sum_{m'=1}^M \sum_{m=1}^\infty \langle \chi_m, \chi_{m'}\rangle B_m(t)\chi_{m'} \\ &= \sum_{m'=1}^M \sum_{m=1}^\infty \delta_{mm'}B_m(t)\chi_{m'} = \sum_{m=1}^M \chi_m B_m(t).\end{aligned} \tag{2.2.10}$$

[11] Compared to Definition 2.2.5, we now require $-a(u,v)$ instead of $\langle u, \mathcal{A}v\rangle$ in the weak formulation, and also different solution and test function spaces.

Including this noise approximation (2.2.10), we can express the last term of (2.2.9) as

$$\langle G\left(\tilde{\rho}(t)\right)\mathcal{P}^M dW(t), \phi_i\rangle = \sum_{m=1}^{M} \langle G(\tilde{\rho}(t))\chi_m, \phi_i\rangle dB_m(t).$$

The choice of the cut-off M is not exactly clear. After applying $G(\tilde{\rho(t)})$ to the truncated process, it will be projected onto $\tilde{V}$ again.

With

$$G(t)_{im} = \langle G(\tilde{\rho}(t))\chi_m, \phi_i\rangle,\ i = 0, \dots, n,\ m = 1, \dots, M,$$
$$dW^M(t) = (dB_m(t))_{m=1}^M,$$

and further denoting the vector of the non-linear term by

$$F(t) = \langle F(\tilde{\rho}(t)), \phi_i\rangle_{i=0}^n,$$

we can finally write the space-discretization and noise-approximation as

$$C d\beta(t) = \left(-A\beta(t) + F(t)\right) dt + G(t) dW^M(t). \tag{2.2.11}$$

Discretizing in Time

For the time-discretization we divide the time interval $[0, T] = [t_0, t_K]$ into K intervals of fixed size Δt. The semi-implicit Euler-Maruyama method is implicit in the linear terms, but explicit in the non-linear term and makes use of the Euler-Maruyama scheme for discretizing the Brownian motion (see also Section 2.1.2). Denoting functions at time $t_k = k\Delta t$ by a subscript k, e.g. $\tilde{\rho}(t_k) = \tilde{\rho}_k$, the time-discretization is the recursion for $k = 0, \dots, K-1$

$$C(\beta_{k+1} - \beta_k) = -A\beta_{k+1}\Delta t + F_k \Delta t + G_k \Delta W_k^M$$

or after rearranging,

$$\beta_{k+1} = (C + A\Delta t)^{-1}\left(C\beta_k + F_k\Delta t + G_k \Delta W_k^M\right).$$

The random increments in time are given by

$$\Delta W_k^M = \left(\int_{t_k}^{t_{k+1}} dB_m(t')\right)_{m=1}^M = (B_m(t_{k+1}) - B_m(t_k))_{m=1}^M = (\sqrt{\Delta t}\zeta_{m,k})_{m=1}^M$$

with i.i.d. $\zeta_{m,k} \sim \mathcal{N}(0, 1)$.

3. Agent-based Model on the Micro-scale

Describing complex systems and phenomena is of interest in order to make predictions, test scenarios, prevent unwanted situations and get new insights into the system. Micro-scale models are models that describe the system on the smallest scale and are thus often the most accurate and detailed but also complicated models. Usually these models define the behavior of a large number of discrete entities, called particles or agents.

Particle-based models have their background in the physics and chemistry literature, whereas agent-based models (ABMs) have their origins and applications in the social sciences, economics, humanities and are much more general. ABMs are computational models that describe the actions and interactions between autonomous entities (called agents) and their environment. The hope is that global patterns emerge from the interplay of the local behaviour of agents [25, 35]. A clear mathematical formulation and model reproducibility is often lacking though [22, 23]. Due to the large body of mathematical theory for particle-based models originating from the Doi and Smoluchowski model [12, 46] and the missing theory for ABMs, we use particle-based models as the basis for our mathematical formulation of an agent-based model. Our ABM definition has similarities to the SIR model for infection spreading [30, 39] and to Brownian agents [45]. In this chapter we closely follow previous work by the author and collaborators [7, 8] but have included more general interaction rules for agents.

Our agent-based model consists of two ingredients: the position dynamics and the interaction rules for agents [7, 8]. The position dynamics of agents are described by Brownian motion with two drift factors. The agents are taking into account the suitability of their environment (drift in the suitability landscape) and also the density of other near-by agents (attraction-repulsion forces). Further agents can interact according to a set of predefined rules whenever they are close in space. They can change their type (e.g. opinion, innovation, information) influenced by the state of other near-by agents. Whilst the type could be a continuous state in general, we consider discrete types only. To be more precise, we model interactions between close-by pairs of agents such that whenever an agent of a certain type A is within a fixed radius of an agent of type B, we have the type change $A + B \to C + B$ at a certain rate. Thus the agent of type A changes its type to C triggered by an interaction with the type B agent who remains unchanged.
This simple class of interaction rules could model for example infection spreading: an agent is infected with a disease (type B) and comes into contact with a susceptible agent (type A) and infects the susceptible agent with a certain rate [5, 39]. Another possible application are opinion dynamics: an agent with a certain opinion B meets an agent with a different opinion A and manages to convince the other agent to change his opinion. In general, we could consider more complicated interaction rules, including the death and birth of agents or the spontaneous type change of an agent. For a list of possibilities framed in terms of chemical reactions, see [14].

We will introduce our agent-based model in Section 3.2, and explain a method for its simulation in Section 3.3. The model is formulated in terms of stochastic processes as introduced in Section 2.1. Later it will turn out that the ABM is computationally very expensive to simulate

and scales badly for an increasing number of agents (see Chapter 5). To make model simulations feasible, model reductions and approximations are needed e.g. by spatial discretizations or by going from individual agents to densities or concentrations of agents. Therefore in Chapter 4, we consider systems of a large number of agents. By making approximations, we arrive at a meso-scale stochastic description of our ABM in terms of a density of agents, which we will therefore call density-based or meso-scale model.

3.1. Motivation: Modeling Innovation Spreading in Ancient Times

Before diving straight into the agent-based model description, we will first give a motivating example of how ABMs can be used to study real-world processes.

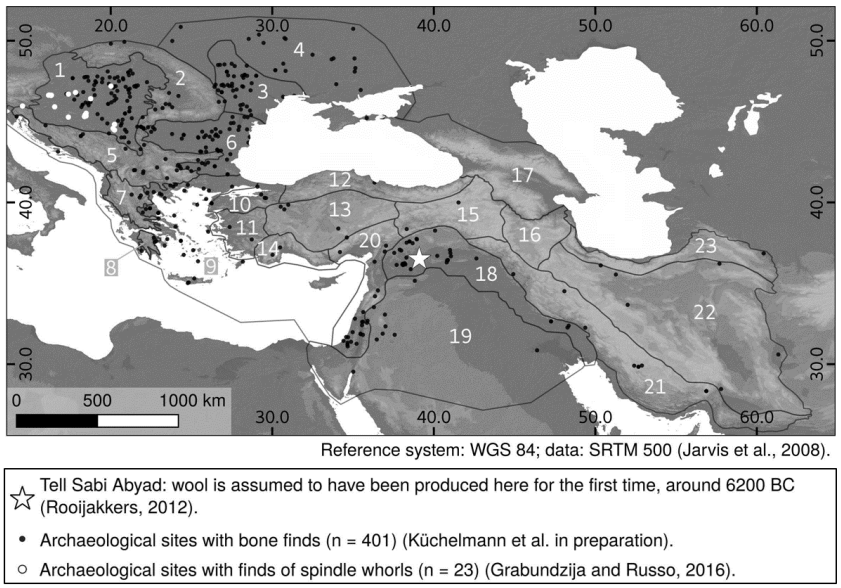


Figure 3.1.: The area of interest for studying the spreading of the wooly sheep with an assumed origin in Tell Sabi Abyad [43], Figure from [8]. The available archaeological bone findings only give evidence and hints as to where ovicaprids (sheep and goats) were farmed. The spindle whorl findings [21] are limited to a small part of the considered area and give suggestions as to where wool or fiber was processed.

Many change processes from ancient times, i.e. the times before there were written records of events, are still largely unclear and debated. Archaeological data is sparse and the datings are uncertain. Thus a reconstruction of historical processes based just on data is often impossible. To give an example, a lot of archaeological research concerns the spreading of farming. One is for example interested in uncovering the possible spreading paths of the wooly sheep from modern-day Syria into Europe between 6000 to 2000 BC [2]. Before the farming of the wool-bearing sheep started around 6200 BC in Tell Sabi Abyad [43], the farming of hairy sheep was common. But the available data about hairy and wooly sheep is not specific enough to deduce possible spreading paths. It is not even possible to distinguish between goats and sheep on the

basis of bone findings. Further, the data about wool-processing tools is limited to a small part of the area of interest as shown in Figure 3.1.

The approach that was taken by the author and collaborators instead [7, 8, 38] is to build an agent-based model on the basis of geographical and geological data in order to simulate and reconstruct possible scenarios. The modeled dynamics can then be studied by performing a sensitivity analysis of the parameters and the model outcome can be compared to the available archaeological data and discussed by experts.

The wooly sheep can be considered as an innovation because its introduction, possibly due to a mutation, replaced the wide-spread herding of hairy sheep. By viewing nomadic groups of sheep herders as agents and modeling the spreading of the innovation amongst them, possible spreading scenarios can be simulated and studied. The simulation of such real-world dynamics is costly, making model reduction techniques a necessity, see Chapter 4. In Section 3.4, we will come back to this setting of innovation spreading and study a simple toy example. This guiding example of innovation spreading will also appear several times in this thesis to make different modeling aspects more concrete.

3.2. Model Formulation

Let us in the following lay out a very general agent-based model that can be used to study many historical and social processes as well as chemical reactions. This model description closely follows the approach in [8] but extended to cover more general interaction rules.

An agent represents a discrete entity such as a person, a group of people or an organization. Each agent is characterized by its type (also called species) and its position. We are following every agent i, $i = 1, \dots, N$ individually and track the evolution in time of its position state and type. The agents' positions are restricted to a given domain $D \subseteq \mathbb{R}^d$. Usually in real-world systems $d = 2, 3$. The type of an agent is denoted by values in $\{1, \dots, N_T\}$, such that there are N_T types in total. In the case of innovation spreading, the type should indicate whether the innovation has been adopted by the agent or not, thus $N_T = 2$.

Then the state of the i^{th} agent at time t is given by

$$(X_i(t),\ Y_i(t)) \in D \times \{1, \dots, N_T\}.$$

Whereas the system state is

$$(X(t), Y(t)) = (X_i(t),\ Y_i(t))_{i=1,\dots,N}$$

with state space $D^N \times \{1, \dots, N_T\}^N$.

Agents are able to move and change their position in the domain $D \subseteq \mathbb{R}^d$ by taking into account their surroundings. Each agent has only local knowledge and can influence other agents only

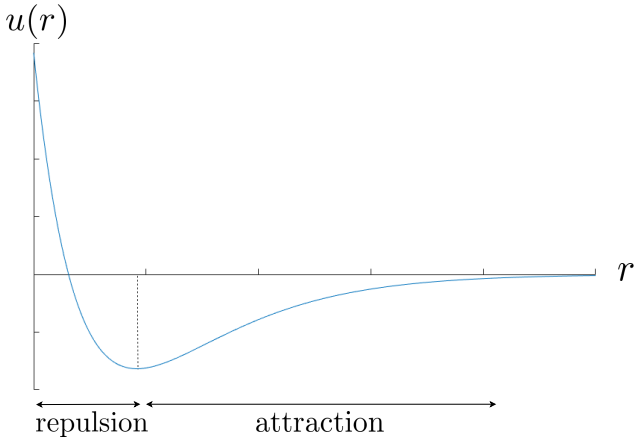


Figure 3.2.: Pair-wise attraction-repulsion potential $u(r)$ depending on the distance r between a pair of agents. If two agents are very close to each other in position space, they are pushed apart. If they are very far from another, they are attracted towards each other. The minimum of the potential stands for the most convenient positioning between a pair of agents.

in his neighborhood. Agents are attracted to near-by regions that are suitable for them and refrain from unsuitable parts of the domain, but they cannot leave the domain (no flux of agents is crossing the boundary). Further, agents tend to group together in space and form clusters, while also keeping some distance from each other in order to avoid spatial overlap. The position dynamics are thus interdependent. We additionally include some randomness in the agents' motion to account for other unknown incentives for positional changes and to allow agents to be explorative or make mistakes in their evaluation of the environment.

Agents change their type according to a set of N_R interaction rules. This happens at a certain rate and whenever they are in proximity of specific other agents. We consider a set of different interaction rules $\{R_r\}$, $r = 1, \dots, N_R$ that are coupled to the agent position dynamics.

3.2.1. Modeling the Agent Position Dynamics

The change for the position $X_i(t) \in D \subseteq \mathbb{R}^d$ of every agent $i = 1, \dots, N$ is governed by the Itô diffusion process

$$dX_i(t) = -\left(\nabla V(X_i(t)) + \nabla U_i(X(t))\right) dt + \sigma dB_i(t), \tag{3.2.1}$$

with $X(t) \in D^N \subseteq \mathbb{R}^{d\times N}$ denoting the positions of the system of agents, $V : D \subseteq \mathbb{R}^d \to \mathbb{R}$ denotes the suitability landscape, $U_i : D^N \subseteq \mathbb{R}^{d\times N} \to \mathbb{R}$ is the i^{th} agents' attraction-repulsion potential with respect to all other agents, $\sigma \in \mathbb{R}$ is a diffusion constant (could in general be type or space-dependent) and $B_i(t)$ denote independent standard Brownian motions in $\mathbb{R}^d$. Every agents' movement is described by this diffusion equation. Thus we have N equations in total, coupled via the attraction-repulsion potential. The diffusion process is independent of the agent types $Y(t)$ though.

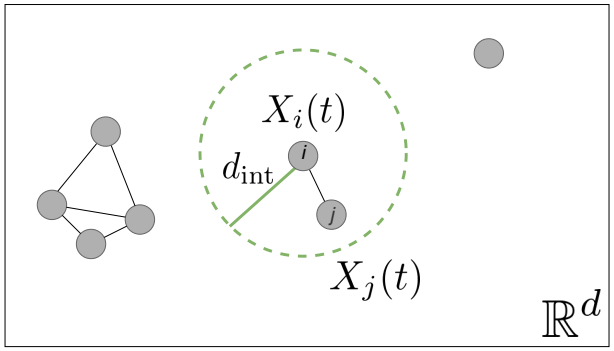


Figure 3.3.: Construction of the contact network based on the agent positions at time t.

The suitability landscape V indicates the attractivity of the environment and gives an incentive to prefer or avoid certain near-by parts of the domain. Valleys of the suitability landscape V correspond to attractive regions and peaks and divides correspond to unsuitable areas that are moreover difficult to overcome.

The attraction-repulsion potential is inspired by interatomic potentials from Physics (e.g. Lennard-Jones potential, Morse potential, Buckingham potential) and drives agents to change their position due to other near-by agents. Attraction between agents occurs whenever agents at long distances are driven towards another, and repulsion appears when agents are forced apart at short distances. Agents are thus searching for an optimal balance between forming clusters of agents on the one hand and distributing in space on the other hand.
The attraction-repulsion felt by agent i due to all other agents $j = 1, \ldots, N,\ j \neq i$ is of the form

$$U_i(X(t)) = \sum_{j=1, j\neq i}^{N} u\left(\|X_i(t) - X_j(t)\|\right),$$

where $\|\cdot\|$ refers to the Euclidean distance. As such it is the sum of the pair-wise attraction-repulsion potentials $u(r)$ between agent i and agent j. There is some minimum of $u(r)$, such that the pair-wise distance $r := \|X_i(t) - X_j(t)\|$ is optimal, see also Figure 3.2. For a smaller respectively larger r, the agents are drawn towards the optimal r and we have repulsion respectively attraction. For $r \to \infty$, no force is felt anymore since $u \to 0$.

3.2.2. Modeling Interaction Rules for Agents

Given the positional movements of agents in the domain, we can construct a network between agents that is changing in time. At time t the network is constructed in the following way: the set of nodes represents the set of agents, and an edge exists between two nodes if the corresponding agents are within a d_{int} radius of each other, see Figure 3.3 for an illustration. The network is

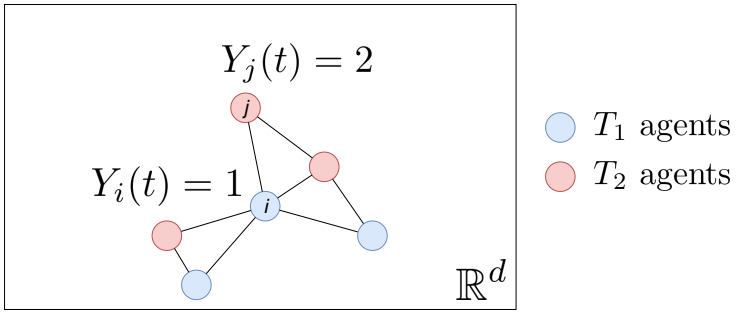


Figure 3.4.: Toy example governed by the interaction rule $R_1 : T_1 + T_2 \to 2\,T_2$. We are considering the interactions between agent i and his connections. Since the interaction rule describes interactions of type T_1 agents with type T_2 agents, agent i can only interact with his three type T_2 contacts. The transition rate function for the i^{th} agent is then $\lambda_i(t) = 3\,\gamma^1_{\text{micro}}$.

fully determined by a time-evolving adjacency matrix $A(t) = (A_{ji}(t))_{i,j=1,\dots,N}$ with entries

$$A_{ji}(t) = \begin{cases} 1 & \text{if } i \neq j,\ \|X_i(t) - X_j(t)\| < d_{\text{int}} \\ 0 & \text{else.} \end{cases}$$

Agents are only interacting with their network contacts. According to the set of N_R interaction rules $\{R_r\},\ r = 1, \dots, N_R$, agents can change their type whenever they are close to each other.

Each rule R_r can be written as the type change

$$R_r :\ T_s + T_{s''} \to T_{s'} + T_{s''} \tag{3.2.2}$$

that happens at the fixed influence rate γ^r_{micro} and conditional on an agent of type T_s being in contact with an agent of type $T_{s''}$, for $s, s', s'' \in \{1, \dots, N_T\}$.
For example when modeling the spreading of an innovation on a network, every agent can only be in one of the two discrete innovation states: T_1 for a non-adopter and T_2 for an adopter of the innovation. Then we are defining one interaction rule R_1: $T_1 + T_2 \to 2\,T_2$, such that adopters pass on the innovation to non-adopters, see additionally Figure 3.4.

We are now interested in describing the type changes for agent $i = 1, \dots, N$. If agent i at time t is of type T_s, $s \in \{1, \dots, N_T\}$, we denote this by $Y_i(t) = s$. Changes for agent i are modeled as Markov jump processes with time-dependent transition rates. The transition rates are changing in time since they depend on the proximity of other agents and their types. The transition rate function $\lambda^r_i(t)$ gives the rate for agent i to change its type according to interaction rule R_r and is proportional to the constant influence rate γ^r_{micro} and to the number of neighbors of agent i that trigger interaction R_r (i.e. the number of agents of type $T_{s''}$ in rule (3.2.2)).

Thus we define the transition rate function for agent i due to interaction rule R_r as

$$\lambda_i^r(t) = \lambda_i^r(A(t), Y(t)) = \gamma_{\text{micro}}^r \sum_{j=1}^{N} A_{ji}(t)\ \mathbb{1}_{\{s''\}}(Y_j(t))\ \mathbb{1}_{\{s\}}(Y_i(t)), \tag{3.2.3}$$

where $\mathbb{1}_B : X \to \{0,1\}$ is the indicator function defined as

$$\mathbb{1}_B(x) := \begin{cases} 1 & \text{if } x \in B \\ 0 & \text{if } x \notin B. \end{cases}$$

Let $\{Y_i(t)\}_{t\geq 0}$ be the type change process of agent i, which can be expressed in terms of Poisson processes as

$$Y_i(t) = Y_i(0) + \sum_{r=1}^{N_R} \mathcal{P}_i^r \left(\int_0^t \lambda_i^r(t')dt' \right) v_r. \tag{3.2.4}$$

The initial type of agent i is $Y_i(0)$, $\mathcal{P}_i^r$ denote i.i.d. unit-rate Poisson processes and the type change vector is denoted by $v = (v_r)_{r=1,\dots,N_R}$ (for R_r as given above $v_r = s' - s$).

3.2.3. Formulation of the Agent System Dynamics

Putting together Equations (3.2.1) and (3.2.4), the coupled agent system equations read

$$\begin{aligned} X_i(t) &= X_i(0) - \int_0^t \left(\nabla V(X_i(t')) + \nabla U_i(X(t')) \right) dt' + \sigma \int_0^t dB_i(t'), \\ Y_i(t) &= Y_i(0) + \sum_{r=1}^{N_R} \mathcal{P}_i^r \left(\int_0^t \lambda_i^r(t')dt' \right) v_r \end{aligned} \tag{3.2.5}$$

for $i = 1, \dots, N$.

These agent system dynamics are coupled in several ways. The position dynamics of all agents are coupled via the attraction-repulsion potentials. And the type change process for each agent i depends on the positions of all agents and on the types of all agents via the time-dependent transition rate functions $\lambda_i^r(t)$ (3.2.3).

Further we want to remark that the modeled dynamics (3.2.5) are memoryless (Markovian) since both the overdamped Langevin equation as well as the Poisson process are memoryless processes.

Since we cannot solve (3.2.5) analytically, we will in the following explain how to discretize and simulate trajectories of the dynamics.

3.3. Simulation Aspects

The goal is to accurately but also efficiently discretize the agent system dynamics (3.2.5) such that we can simulate trajectories of the discretized process. Often one is not just interested in single realizations of the system, but in getting meaningful information about the average dynamics and the deviations from the average. For computing model averages or higher order moments, we need to sample a large enough ensemble of trajectories in order to compute reasonable Monte Carlo estimates of the quantities of interest. Thus we want to keep the computational cost of a single realization to be low.

Our ABM is a coupled system, the movements of agents are coupled via the attraction-repulsion potential. One therefore needs to simulate the position dynamics of all agents simultaneously. But also the type changes of agents are coupled to the position dynamics. The interactions have to be simulated simultaneously for all agents and after the position dynamics. The most straightforward approach is therefore to simulate both dynamics in parallel, to simulate the trajectories of the position dynamics using an Euler-Maruyama discretization of the SDE (3.2.1) and to make a time-discrete approximation for the type change processes (3.2.4).

We discretize time by $t_k = k\Delta t$, for $k = 0, \dots, K-1$ and with a sufficiently small time step Δt. After setting all parameters and the initial conditions $(X(0), Y(0)) = (X_0, Y_0)$, the simulation approach consists of the following steps.

For each $k = 0, \dots, K-1$:

(a) Position Dynamics.

For each agent $i = 1, \dots, N$, we advance the agent positions by

$$X_i(t_{k+1}) = X_i(t_k) - \left(\nabla V(X_i(t_k)) + \nabla U_i(X(t_k))\right)\Delta t + \sigma\sqrt{\Delta t}\xi$$

with i.i.d. $\xi \sim \mathcal{N}(0,1)$ in $\mathbb{R}^d$.

We need to compute the distances between pairs of agents in each time step. On the one hand this is needed for building the attraction-repulsion potential, which depends on the pairwise distance between agents. We can introduce a cut-off radius for the attraction repulsion potential since $u \to 0$ for far-away agents, and thereby neglect the attraction-repulsion between far-away agents. On the other hand, we need to construct a neighbourhood graph with radius d_{int} in order to check whether two agents are interacting (for step (b) below). Simply checking all pair-wise distances is a very expensive computation scaling like $\mathcal{O}(N^2)$ for N agents. But there are some work-arounds. One can make the computations substantively less expensive by using e.g. k-d trees [3].

(b) Interaction Rules.

We simulate type changes of agents by a time-discretization of the Poisson processes with the same Δt time steps as for the discretization of the position dynamics. During each interval $[t_k, t_{k+1})$, we check all possible type change events by iterating over all agents $i = 1, \ldots, N$ and rules $R_r,\ r = 1, \ldots N_R$. Assuming the transition rate functions are approximately constant over the time interval, the probability for agent i to change its type due to interaction rule R_r is

$$1 - \exp(-\lambda_i^r(t_k)\Delta t).$$

This means that we check whether a type change event happens in a time interval by drawing $\theta \sim \mathcal{U}(0,1)$. If $\theta \leq 1 - \exp(-\lambda_i^r(t_k)\Delta t)$ the event is accepted, else no type change event takes place and we draw another random number in order to consider the next event.

The choice of Δt for step (b) is a compromise between accuracy and efficiency. On the one hand, we want the step size Δt to be not too small, such that the algorithm does not become costly. For a too small Δt, there are many time intervals during which no jump event takes place. On the other hand, we want to keep Δt small in order to approximate the type change processes well. For a too large Δt, we neglect the immediate effect of a type change on other agents [17]. Besides, a continuous-time process is better approximated using a small Δt.

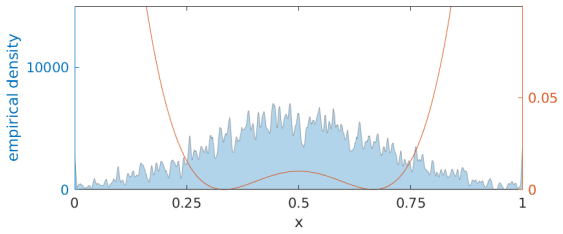
Instead of a time-discrete approximation of the type change processes, another possibility is to sample the type changes statistically exact and continuously in time. Since the transition rate functions are changing in time and even in between events, we can make use of the Temporal Gillespie algorithm [47, 7], which is similar to the algorithm introduced in Section 2.1.1 for inhomogeneous Poisson processes. The Temporal Gillespie algorithm predicts in this setting the waiting time until the next type change and the specific type change that will happen. The coupling of the Temporal Gillespie algorithm with an Euler-Maruyama discretization for the position dynamics has been outlined in [7].

3.4. Numerical Example: Innovation Spreading in a Double Well Landscape

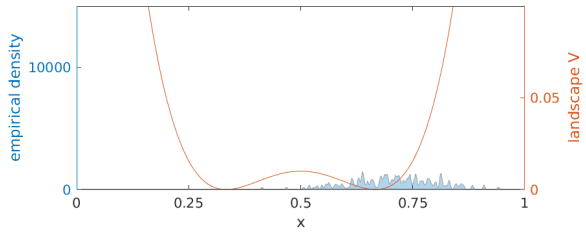
We are in the following studying a toy example of $N = 3000$ agents diffusing in a double well landscape

$$V(x) = \left(3.6\,(x - 0.5)^2 - 0.1\right)^2$$

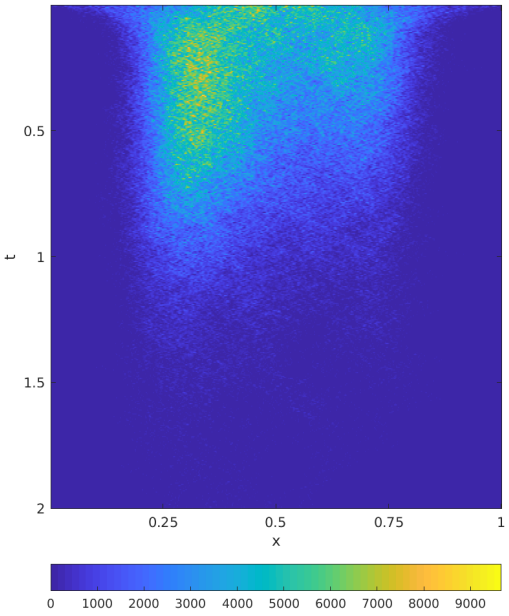
on $D = [0, 1]$. The double well landscape is characterized by two minima centered at $x = \frac{1}{3}$ and $x = \frac{2}{3}$, corresponding to the most suitable areas for agents, and a barrier between the two wells. On top of that, we choose a few agents at time $t = 0$ to be adopters of an innovation that they are passing on to other near-by agents. Agents can either be adopters of the innovation (type T_2) or non-adopters of the innovation (type T_1). Whenever a type T_1 agent is in contact with a type T_2 agent, he adopts the innovation at the fixed rate $\gamma^1_{\text{micro}} = 0.5$ and for all times. Thus



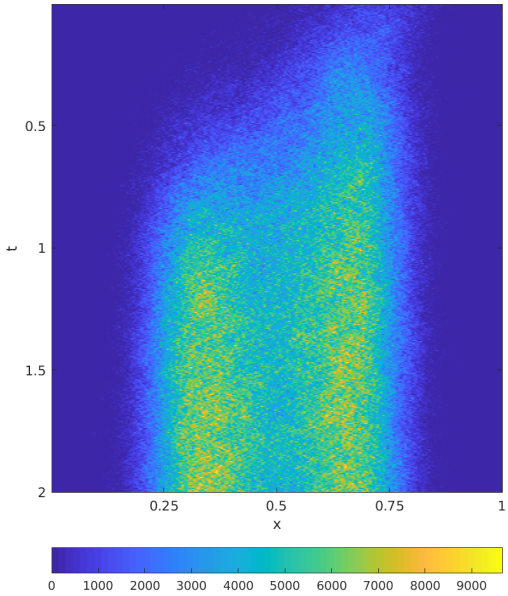
(a) Initial empirical density of non-adopters.



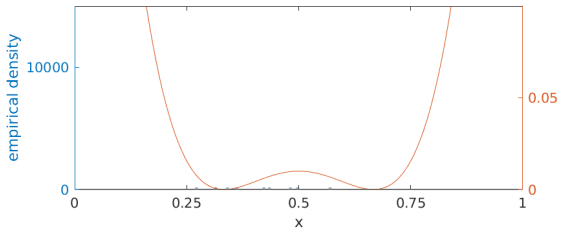
(b) Initial empirical density of adopters.



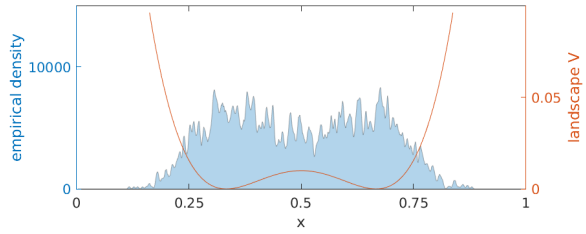
(c) Evolving empirical density of non-adopters plotted using a heat map.



(d) Evolution of the empirical density of adopters plotted using a heat map.



(e) Final empirical density of non-adopters.



(f) Final empirical density of adopters.

Figure 3.5.: Guiding example of agents diffusing in a double well and spreading of an innovation. Initially, 200 agents are adopters of the innovation with their positions sampled from $\mathcal{N}(0.7, 0.01)$. At time $t = 0$ 2800 agents have not yet adopted the innovation, their positions are distributed according to $\mathcal{N}(0.5, 0.04)$. Further model and numerical parameters: $\sigma = 0.25$, $d_{\text{int}} = 0.001$, $\Delta t = 0.005$.

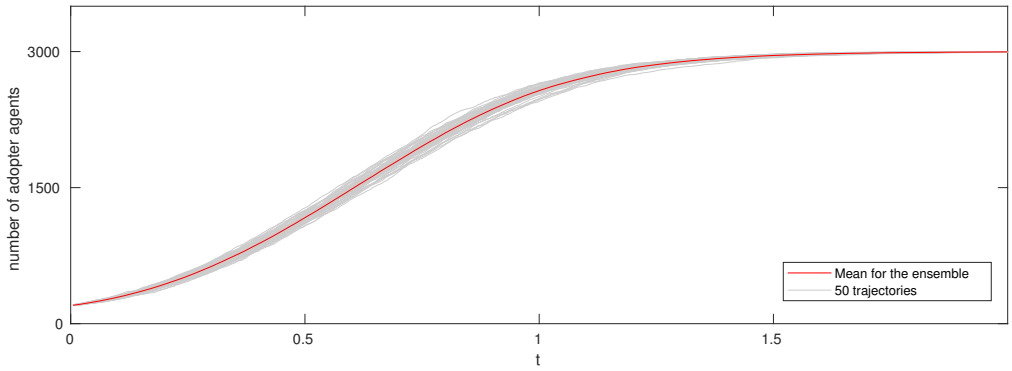


Figure 3.6.: For an ensemble of 50 simulations we look at observables of the system such as the number of adopters in time t.

the only interaction rule is

$$R_1 : \; T_1 + T_2 \to 2\, T_2.$$

The attraction-repulsion potential U_i is disregarded in this example. The agent dynamics as outlined above are described by a system of coupled stochastic processes (3.2.5) and can be simulated on the basis of the approach explained in Section 3.3.

This very simple model already exhibits interesting patterns. One simulation (or realization) of the process is shown in Figure 3.5. We are plotting the evolution of the empirical agent density for $t \in [0, 2]$ for both agent types. The empirical agent density is constructed by placing a unit mass at each agents' position, i.e. we sum over all agents and at each agents' position we place a function centered at the position and that integrates to one (e.g. a hat function or a Gaussian function).
We observe the following dynamics on the global scale. At time $t = 0$ the innovation starts spreading in the well centered at $x = \frac{2}{3}$. The agent densities quickly distribute near the attractive centers of the two wells. It takes some time until the innovation reaches the other well centered at $x = \frac{1}{3}$. But as soon as an adopter agent crosses the barrier for the first time, the innovation quickly gets adopted by all agents in the other well. Innovation spreading inside the wells is fast, since the agents are closely packed. But the spreading across the barrier takes a long time due to the metastable agent movement (recall the Overdamped Langevin equation in Section 2.1.3). At the final time $t = 2$ all agents are of type T_2. The empirical agent densities are very noisy, since the diffusion and interactions of agents are described by stochastic processes. Since there are only 3000 agents, the noisiness is still dominant on the global scale.

We are also interested in studying an ensemble of trajectories to get a full picture of the system dynamics and its variations. As an observable of the system we study the evolution of the amount of T_2 agents in time. In Figure 3.6, we are plotting this observable for each realization of the ensemble as well as in mean. We observe that as time advances, more and more agents are adopting the innovation. Even though the behaviour of each individual agent is very random

and differs completely between each realization, we can observe that the emergent dynamics are only slightly random with small deviations from the mean.

Using more data and a two-dimensional domain covering Asia and Europe, we can extend this guiding example to consider the motivating question from the beginning of this chapter: "What could have been possible spreading paths of the wooly sheep from modern-day Syria into Europe between 6000 to 2000 BC?" This has been the topic of research in [7, 8, 38]. It was shown that the modeled spreading paths are strongly influenced and dominated by the mobility dynamics. Even more, one can conclude that metastability in the diffusion process induces metastability in the innovation spreading. We have already observed this phenomenon in our toy example.
To understand the historical process, one should produce and study a large ensemble of trajectories of the modeled dynamics. Only in that way, the role of the inherent model stochasticity can be interpreted. One can speculate whether the true prehistorical spreading path correlates with one of the trajectories of the modeled process. Instead one should rather view the modeled spreading path as a hypothesis which has to be discussed and evaluated with the help of additional expert knowledge.

4. Towards a Density-based Description on the Meso-scale

The agent-based model introduced in Chapter 3 describes the stochastic motion of different types of discrete agents and their local interactions according to a set of predefined rules. It is formulated in terms of each individual agent, which becomes computationally too expensive to simulate for large populations. However we do not intend to study every single agent trajectory. Instead we want to analyse the global, emergent dynamics of the whole population. To do this more efficiently, the model complexity can be reduced.

One of our goals is therefore to find a valid and adequate approximation to the agent-based description for large populations of agents on the meso-scale. In this chapter we will derive such a meso-scale model formulated as a system of stochastic PDEs propagating number densities of different types of agents (based on [31, 10]). When changing our view point from individual agents in the ABM to agent densities as depicted in Figure 4.1, agents become indistinguishable among their type and we lose the individual agent labels.
In general for N agents and a spatially inhomogeneous system, the agent-based formulation can be considered as the most accurate model. But for a larger number of agents and many interactions the SPDE formulation is a good approximation and much faster to simulate. In this meso-scopic model, stochasticity still emerges from the systems' inherent randomness. And for the number of agents approaching infinity, stochastic effects can be neglected and thus a PDE model is appropriate.

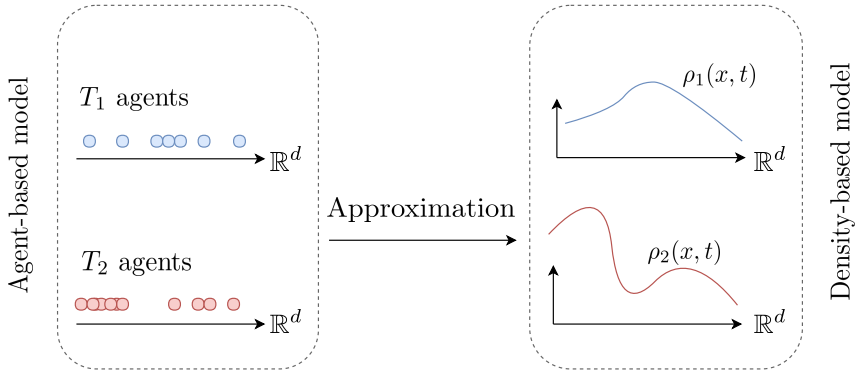


Figure 4.1.: Model reduction from the agent-based model formulated in terms of N discrete agents to the density-based model. The density-based model describes the stochastic evolution of agent number densities $\rho_s(x,t)$ for each type $s = 1, \dots, N_T$.

In Section 4.1 we start by proposing the reduced model in the form of a system of SPDEs. In Section 4.2 we show that it indeed is a consistent reduction of the ABM dynamics before tackling its efficient simulation in Section 4.3. Thereby we will build on the theory introduced in Section 2.2.

4.1. Formulation of the Density-based Model

Before deriving the density-based model as an approximation on the meso-scale in Section 4.2, we will state and explain its features.

We are considering a model describing the stochastic evolution of stochastic agent number densities (or number concentrations) for each agent type $s = 1, \dots, N_T$. The (stochastic) agent number density

$$\rho_s(x,t) : D \times [0,T] \mapsto \mathbb{R}_{\geq 0},$$

is defined on the domain $D \subseteq \mathbb{R}^d$, the time interval $[0,T]$ and with probability space $(\Omega, \mathcal{F}, \mathbb{P})$[1]. Integrating the number density over the domain yields the number of agents N_s of type T_s.

The densities $(\rho_1(x,t), \dots, \rho_{N_T}(x,t))$ evolve due to diffusion and drift in the suitability landscape (neglecting the attraction-repulsion potential for simplicity) and because of the set of interaction rules. The temporal changes of $\rho_s(x,t)$ for $s = 1, \dots, N_T$ are given by the stochastic partial differential equation,

$$\frac{\partial \rho_s(x,t)}{\partial t} = \mathcal{D}\rho_s(x,t) + \mathcal{I}\rho_s(x,t) \tag{4.1.1}$$

with stochastic diffusion operator $\mathcal{D}$ and stochastic interaction operator $\mathcal{I}$. For a fixed sample $\omega \in \Omega$, the agent density is a realization of a stochastic process solving the SPDE (4.1.1).

The diffusion operator of Equation (4.1.1) is given by [31, 10]

$$\mathcal{D}\rho_s(x,t) = \frac{\sigma^2}{2}\Delta\rho_s(x,t) + \nabla \cdot (\nabla V(x)\rho_s(x,t)) + \sigma\nabla \cdot (\sqrt{\rho_s(x,t)}Z_s^{\mathcal{D}}) \tag{4.1.2}$$

with $\sigma \in \mathbb{R}$, suitability landscape $V : \mathbb{R}^d \mapsto \mathbb{R}$ and with $Z_s^{\mathcal{D}} = \left(Z_{s,1}^{\mathcal{D}}, \dots, Z_{s,d}^{\mathcal{D}}\right)$ denoting a d-dimensional vector of space-time white noise for the diffusion, i.e.

$$\mathbb{E}\left(Z_{s,j}^{\mathcal{D}}(x,t)Z_{s',j'}^{\mathcal{D}}(x',t')\right) = \delta_{jj'}\delta_{ss'}\delta(x-x')\delta(t-t').$$

The diffusive part of the SPDE evolves a number density of many agents and is responsible for the diffusive transport in space with drift in the suitability landscape $V(x)$. The number density is something different than a probability density, still we can point out some similarities to the Fokker-Planck equation (2.1.3). The Fokker-Planck equation can be interpreted as an equation describing the evolution of a density of infinitely many non-interacting particles that are diffusing with drift in a potential. Here we have a large, but finite number of diffusing particles and an additional random forcing (the first two terms on the right-hand side of (4.1.2) are exactly the Fokker-Planck equation). Since we have finitely many agents, our model description is still inherently random.
The noise term is non-linear and multiplicative. Due to this term it is not clear yet whether a

[1] Actually the number density should be $\rho_s(\omega, x, t)$ but we never write the dependence on $\omega \in \Omega$. The agent number density is random since it solves a stochastic PDE with STWN terms.

solution exists and is unique. Some first research in this direction can be found in [16]. Applying a divergence operator onto space-time white noise formally does not make sense since we cannot differentiate STWN in space. But when considering a weak formulation of the SPDE, the divergence operator can be moved to the test functions.

Moving to the interaction part of the SPDE system (4.1.1), the agent densities are interacting according to a set of N_R rules [31]

$$\mathcal{I}\rho_s(x,t) = \sum_{r=1}^{N_R} \nu_s^r \left(a^r(\rho(x,t)) + \sqrt{a^r(\rho(x,t))} Z_r^{\mathcal{I}} \right). \tag{4.1.3}$$

Space-time white noise for the r^{th} interaction is denoted by $Z_r^{\mathcal{I}}$ with covariance

$$\mathbb{E}\left(Z_r^{\mathcal{I}}(x,t) Z_{r'}^{\mathcal{I}}(x',t') \right) = \delta_{rr'}\delta(x-x')\delta(t-t').$$

The coefficient ν_s^r describes the discrete number change of the type T_s agents involved in the r^{th} interaction rule[2], $a^r(\rho(x,t))$ is the transition rate function for densities due to the r^{th} interaction rule. The idea of the transition rate function for agent densities is similar to the transition rate functions for the ABM (3.2.3). The rate function depends on the local amount of the two types of agents taking part in the interaction. The more agents of each of the two types, the more interactions are happening. Also for a larger rate constant influence rate γ_{meso}^r (units of γ_{meso}^r are volume × inverse time), more interactions are happening per time.

To get a better understanding of these coefficients and functions, let us return to our innovation spreading example: agents of type T_1 and T_2 are interacting according to the rule $R_1 : T_1 + T_2 \to 2\, T_2$. Since for each interaction the number of species T_1 decreases by one agent and the number of species T_2 increases by one agent, we have $\nu_1^1 = -1,\ \nu_2^1 = 1$. The transition rate function for the interaction between two agent densities is proportional to the density of each and the rate γ_{meso}^1 such that

$$a^1(\rho(x,t)) = \gamma_{\text{meso}}^1 \rho_1(x,t)\rho_2(x,t)$$

in this example.

In order to derive the interaction part of the SPDE (4.1.3), the Poisson random variable for interactions in the agent-based description will be replaced by a Gaussian random variable with the same mean and variance. Further the interaction parameters d_{int} and γ_{micro}^r are aggregated to the meso-scale influence rate γ_{meso}^r. This approximation is only valid for large[3] $a(\rho(x,t))$, and is closely related to the approximation that has been done for well-mixed systems of interacting species leading to the chemical Langevin equation (CLE) [19]. Since Equation (4.1.3) includes spatial information and does not describe a well-mixed system, it can be viewed as a spatial

[2] In the chemistry literature, ν_s^r is called the stoichiometric coefficient of type T_s due to the r^{th} reaction.

[3] To be more precise, we assume the transition rate function to be large in each grid cell after discretization of the system equations.

extension of the CLE.

As we have now described the system of SPDEs, there are some further points worth noting. We can observe that the diffusive part (4.1.2) conserves the number of agents of each type because of the divergence operator form and assuming no-flux boundary conditions. Hence it is responsible for the transport of the density in space. The interaction part (4.1.3) of the SPDE shifts the agent number density between the different agent types but conserves the overall number of agents of all types, i.e. $\sum_{s=1}^{N_T} N_s = N$ is conserved.

We can also have a closer look at how the noise terms scale for increasing agent numbers, i.e. larger values of the density $\rho_s(x,t)$. The noise terms are scaled by a square-root factor of the agent density, whereas all the other (deterministic) terms are scaled by the density. Thus for the number of agents approaching zero, the noise dominates the SPDE. For the number of agents going to infinity, the noise terms become unimportant and could be neglected. Thus the SPDE would turn into a PDE.

For the SPDE to be solvable, boundary conditions and initial conditions have to be specified. The domain boundary should not be crossed by agents, thus we require a no-flux boundary condition (Neumann). Then the SPDE reads for agent types $s = 1, \dots, N_T$,

$$\begin{aligned} \frac{\partial \rho_s(x,t)}{\partial t} &= \mathcal{D}\rho_s(x,t) + \mathcal{I}\rho_s(x,t) && \text{on } D \times [0,T] \\ \nabla \rho_s(x,t) \cdot \hat{u} &= 0 && \text{on } \delta D \times [0,T] \\ \rho_s(x,0) &= \rho_{s,0}(x) && \text{on } D \times \{0\} \end{aligned} \tag{4.1.4}$$

with the stochastic operators $\mathcal{D}$ and $\mathcal{I}$ as defined above and $\hat{u}$ denoting the unit outer normal to δD. The system of SPDEs for the N_T species is coupled via the interaction terms, more specifically the transition rate functions. The initial data $\rho_{s,0}(x)$ needs to be non-negative.

We further note that the analysis of the well-posedness and existence of solutions to this SPDE system is not investigated enough [16]. In this thesis though, we are concerned with a discretization of the SPDE system. In Section 4.3 we will explain how to numerically sample realizations of the discretized system of SPDEs.

Remark. We can extend the system of SPDEs to include attraction forces between pairs of agents at long ranges and repulsion forces at short ranges. Denoting the attraction-repulsion potential between two agents at positions x and y by $u(\|x-y\|)$, the SPDE is extended by one term [10]

$$\begin{aligned} \frac{\partial \rho_s(x,t)}{\partial t} =& \frac{\sigma^2}{2}\Delta\rho_s(x,t) + \nabla\cdot(\nabla V(x)\rho_s(x,t)) + \nabla\cdot\left(\rho_s(x,t)\int_D\left(\sum_{s'}\rho_{s'}(y,t)\right)\nabla u(\|x-y\|)dy\right) \\ &+ \sigma\nabla\cdot(\sqrt{\rho_s(x,t)}Z_s^{\mathcal{D}}) + \sum_{r=1}^{N_R}\nu_s^r\left(a^r(\rho(x,t)) + \sqrt{a^r(\rho(x,t))}Z_r^{\mathcal{I}}\right). \end{aligned}$$

The additional term models the diffusion of the agent density ρ_s in the aggregated attraction-repulsion potential of the density of all agents $\sum_{s'} \rho_{s'}(y,t)$.
Including this term, the diffusing densities are all coupled to each other. For simplicity, we will later numerically solve Equation (4.1.4) without attraction-repulsion effects.

4.2. Model Reduction on the Meso-scale: From Agent-based to Density-based

We have seen the formulation of the reduced density-based model (the system of SPDEs). Let us in the following show how to derive the system of SPDEs from the agent-based dynamics.
The agent-based model describes and tracks the dynamics of each individual agent. The position dynamics of agents are described by Itô diffusion SDEs (3.2.1) that are coupled to Markov jump processes (3.2.4) for the type changes of agents. However to derive a description in terms of an agent density, we can first derive a stochastic PDE modeling the diffusion of the agent density based on [10], where for the moment the agent types are not important. In addition to the derivation in [10], we also include a suitability landscape $V(x)$ in our model that exhibits a drift onto the agents' motion. Further, we are extending the derivation by some calculations and explanations that were missing. As a next step, in Section 4.2.2 we will derive a model for the interactions between agents [31, 4, 19].

4.2.1. Diffusion of an Agent Density

In the ABM, each agent $i = 1, \dots, N$ diffuses and changes its position in the suitability landscape V (neglecting the attraction-repulsion potential) modeled by the Itô diffusion

$$dX_i(t) = -\nabla V(X_i(t))dt + \sigma dB_i(t), \tag{4.2.1}$$

with i.i.d. standard Brownian motion $B_i(t)$ in $\mathbb{R}^d$ such that

$$\mathbb{E}\left(\frac{dB_i^m(t)}{dt}\frac{dB_j^l(t')}{dt'}\right) = \delta_{ml}\delta_{ij}\delta(t-t'). \tag{4.2.2}$$

A more detailed description of the dynamics can be found in Section 3.2.1.

For distinct agents, whose position dynamics are given by the SDE, we can write down an empirical density at some time t by summing Dirac deltas placed at each agents' position. The Dirac deltas $\delta(x-z)$ can be intuitively thought of as functions that are zero apart from at z, and that integrate to one. For discrete agents $i = 1, \dots, N$ at positions $X_i(t) \in D \subseteq \mathbb{R}^d$, we sum the single agent densities, [4] $\rho_i(x,t)$ to get the agent number density at time t

$$\rho(x,t) = \sum_{i=1}^{N} \rho_i(x,t) = \sum_{i=1}^{N} \delta(x - X_i(t)).$$

[4] The single agent density $\rho_i(x,t)$ is the density of one agent, i.e. a Dirac delta distribution placed at the location of agent i. This should not be confused with $\rho_s(x,t)$, the number density of all agents of species s.

$\rho(x,t)$ has to be interpreted in the sense of distributions. The agent number density integrates to the number of agents $\int_D \rho(x,t)dx = N$ for all t, since by definition of the Dirac delta distribution $\int_D \delta(x - X_i(t))dx = 1$.

In this section, we are deriving an approximate model for the diffusion of many agents consisting of the following steps: First, we will transform the SDE (4.2.1) for agent i into an SPDE describing the temporal evolution of $\rho_i(x,t)$, i.e. of a Dirac Delta function placed at the position of agent i. Second, we will sum the resulting SPDEs for each agent i with the aim of arriving at an SPDE in terms of the number density $\rho(x,t) = \sum_{i=1}^{N} \rho_i(x,t)$. The noise forcing of the SPDE cannot simply be rewritten in terms of $\rho(x,t)$, therefore it will be approximated by a different noise term with the same mean and covariance. Last, we assume N to be large, such that we can replace $\rho(x,t)$ by a number density.

First Step. For the first step in the derivation we need two ingredients: Using the definition of the Dirac delta distribution we have for any test function $\phi \in C_0^\infty(D)$ that

$$\int_D \rho_i(x,t)\phi(x)dx = \phi(X_i(t)).$$

Further we are using the Itô Formula (see Theorem 2.1.5) to find the differential of a function of the stochastic process $\{X_i(t)\}_t$. For our SDE (4.2.1) and any twice-differentiable function $f(x)$, the Itô Formula gives

$$df(X_i(t)) = \left(-\nabla V(X_i(t)) \cdot \nabla f(X_i(t)) + \frac{\sigma^2}{2}\, \Delta f(X_i(t))\right) dt + \sigma \nabla f(X_i(t)) \cdot dB_i(t).$$

Combining the two ingredients we get

$$\frac{df(X_i(t))}{dt} = \int_D \rho_i(x,t)\Big(- \nabla V(x) \cdot \nabla f(x) + \frac{\sigma^2}{2}\, \Delta f(x) + \sigma \nabla f(x) \cdot \frac{dB_i(t)}{dt}\Big)dx.$$

Using integration by parts and that test functions have compact support,

$$\frac{df(X_i(t))}{dt} = \int_D f(x)\left(\nabla \cdot (\nabla V(x)\rho_i(x,t)) + \frac{\sigma^2}{2}\Delta\rho_i(x,t) - \sigma\nabla \cdot \left(\rho_i(x,t)\frac{dB_i(t)}{dt}\right)\right)dx. \quad (4.2.3)$$

Then differentiating $f(X_i(t)) = \int_D \rho_i(x,t)f(x)dx$ by t,

$$\frac{df(X_i(t))}{dt} = \int_D \frac{\partial \rho_i(x,t)}{\partial t} f(x)dx$$

and equating with equation (4.2.3), we get

$$\int_D \frac{\partial \rho_i(x,t)}{\partial t} f(x)dx = \int_D f(x)\left(\nabla \cdot (\nabla V(x)\rho_i(x,t)) + \frac{\sigma^2}{2}\Delta\rho_i(x,t) - \sigma\nabla \cdot \left(\rho_i(x,t)\frac{dB_i(t)}{dt}\right)\right)dx.$$

By the fundamental lemma of calculus of variations and since f is an arbitrary test function, we arrive at the following SPDE for the single agent density $\rho_i(x,t)$,

$$\frac{\partial \rho_i(x,t)}{\partial t} = \nabla \cdot (\nabla V(x)\rho_i(x,t)) + \frac{\sigma^2}{2}\Delta\rho_i(x,t) - \sigma\nabla \cdot \left(\rho_i(x,t)\frac{dB_i(t)}{dt}\right). \tag{4.2.4}$$

Second Step. The goal is to find a closed-form equation for the density of all agents

$$\rho(x,t) = \sum_i \rho_i(x,t).$$

Therefore we sum (4.2.4) over $i = 1,\dots,N$ and by linearity of the differentiation we have

$$\frac{\partial \rho(x,t)}{\partial t} = \nabla \cdot (\nabla V(x)\rho(x,t)) + \frac{\sigma^2}{2}\Delta\rho(x,t) - \sigma\sum_{i=1}^{N}\nabla \cdot \left(\rho_i(x,t)\frac{dB_i(t)}{dt}\right). \tag{4.2.5}$$

Since the noise term (the last term) cannot be turned into a term just depending on the agent density, the idea is to approximate it by a different noise forcing that has the same covariance and mean function but depends only on $\rho(x,t)$ [10]. Investigating the original noise term of (4.2.5)

$$\xi(x,t) \coloneqq -\sigma\sum_{i=1}^{N}\nabla \cdot \left(\rho_i(x,t)\frac{dB_i(t)}{dt}\right),$$

one can show that it has zero mean and covariance function

$$\begin{aligned}
\mathbb{E}\left(\xi(x,t)\xi(y,t')\right) &= \sigma^2\mathbb{E}\left(\sum_{i=1}^{N}\nabla_x \cdot \left(\rho_i(x,t)\frac{dB_i(t)}{dt}\right)\sum_{j=1}^{N}\nabla_y \cdot \left(\rho_j(y,t')\frac{dB_j(t')}{dt'}\right)\right)\\
&= \sigma^2\sum_{i=1,j=1}^{N}\mathbb{E}\left(\left(\nabla_x\rho_i(x,t)\cdot\frac{dB_i(t)}{dt}\right)\left(\nabla_y\rho_j(y,t')\cdot\frac{dB_j(t')}{dt'}\right)\right)\\
&= \sigma^2\delta(t-t')\sum_{i=1}^{N}\nabla_x\rho_i(x,t)\cdot\nabla_y\rho_i(y,t)\\
&= \sigma^2\delta(t-t')\sum_{i=1}^{N}\nabla_x\cdot\nabla_y(\rho_i(x,t)\rho_i(y,t))\\
&= \sigma^2\delta(t-t')\nabla_x\cdot\nabla_y(\delta(x-y)\rho(x,t)).
\end{aligned} \tag{4.2.6}$$

Here we used the covariance for Brownian motion (4.2.2) and in the last line we made use of the identity

$$\rho_i(x,t)\rho_i(y,t) = \delta(x-y)\rho_i(x,t).$$

We will now show that the noise term

$$\tilde{\xi}(x,t) \coloneqq \sigma\nabla \cdot \left(Z(x,t)\sqrt{\rho(x,t)}\right)$$

has the same covariance as $\xi(x,t)$, where $Z(x,t)$ denotes a d-dimensional vector of space-time white noise with

$$\mathbb{E}(Z_j(x,t)Z_{j'}(y,t')) = \delta_{jj'}\delta(x-y)\delta(t-t').$$

For the new noise forcing we have the covariance function

$$\begin{aligned}
\mathbb{E}\left(\tilde{\xi}(x,t)\tilde{\xi}(y,t')\right) =& \sigma^2\mathbb{E}\left(\nabla_x\cdot\left(Z(x,t)\sqrt{\rho(x,t)}\right)\nabla_y\cdot\left(Z(y,t')\sqrt{\rho(y,t')}\right)\right)\\
=& \sigma^2\mathbb{E}\left(\left(\sqrt{\rho(x,t)}\nabla_x\cdot Z(x,t)\right)\left(\sqrt{\rho(y,t')}\nabla_y\cdot Z(y,t')\right)\right)\\
&+\sigma^2\mathbb{E}\left(\left(\sqrt{\rho(x,t)}\nabla_x\cdot Z(x,t)\right)\left(Z(y,t')\cdot\nabla_y\sqrt{\rho(y,t')}\right)\right)\\
&+\sigma^2\mathbb{E}\left(\left(Z(x,t)\cdot\nabla_x\sqrt{\rho(x,t)}\right)\left(\sqrt{\rho(y,t')}\nabla_y\cdot Z(y,t')\right)\right)\\
&+\sigma^2\mathbb{E}\left(\left(Z(x,t)\cdot\nabla_x\sqrt{\rho(x,t)}\right)\left(Z(y,t')\cdot\nabla_y\sqrt{\rho(y,t')}\right)\right).
\end{aligned}$$

To continue further, we need some properties of the Dirac delta distribution and the covariance for derivatives of the space-time white noise [1]

$$\begin{aligned}
\mathbb{E}\left(\frac{\partial Z(x,t)}{\partial x_k}Z(y,t')\right) &= \delta(t-t')\frac{\partial}{\partial x_k}\delta(x-y)\\
\mathbb{E}\left(\frac{\partial Z(x,t)}{\partial x_k}\frac{\partial Z(y,t')}{\partial y_l}\right) &= \delta(t-t')\frac{\partial^2}{\partial x_k\partial y_l}\delta(x-y).
\end{aligned}$$

Equipped with this, we can finish the calculation

$$\begin{aligned}
\mathbb{E}\left(\tilde{\xi}(x,t)\tilde{\xi}(y,t')\right) = \sigma^2\delta(t-t')\Big(&-\sqrt{\rho(x,t)\rho(y,t)}\nabla_x\cdot\nabla_y\delta(x-y)\\
&+\nabla_x\delta(x-y)\cdot\sqrt{\rho(x,t)}\nabla_y\sqrt{\rho(y,t)}\\
&+\nabla_y\delta(x-y)\cdot\sqrt{\rho(y,t)}\nabla_x\sqrt{\rho(x,t)}\\
&+\delta(x-y)\nabla_x\sqrt{\rho(x,t)}\cdot\nabla_x\sqrt{\rho(x,t)}\Big).
\end{aligned}\tag{4.2.7}$$

Since we are dealing with generalized stochastic processes, one needs to multiply the covariance by test functions $f(x)$, $g(y) \in C_0^\infty(D)$ and integrate in order to show that the two covariances for ξ and $\tilde{\xi}$ (equation (4.2.6) and (4.2.7) respectively) agree. It is then enough to show that $\forall f,g \in C_0^\infty(D)$,

$$\begin{aligned}
&\int_D\int_D \nabla_x\cdot\nabla_y(\delta(x-y)\rho(x,t))f(x)g(y)dx\,dy \overset{!}{=} \int_D\int_D\Big(-\sqrt{\rho(x,t)\rho(y,t)}\nabla_x\cdot\nabla_y\delta(x-y)\\
&+\nabla_x\delta(x-y)\cdot\sqrt{\rho(x,t)}\nabla_y\sqrt{\rho(y,t)}+\nabla_y\delta(x-y)\cdot\sqrt{\rho(y,t)}\nabla_x\sqrt{\rho(x,t)}\\
&+\delta(x-y)\nabla_x\sqrt{\rho(x,t)}\cdot\nabla_x\sqrt{\rho(x,t)}\Big)f(x)g(y)dx\,dy.
\end{aligned}\tag{4.2.8}$$

The calculations can be found in the Appendix A.

Finally, we can approximate the original noise forcing by the new noise, based on them sharing the same mean and covariance functions.

Third Step. The SPDE now only depends on $\rho(x,t)$ and not on $\rho_i(x,t)$, and reads as follows

$$\frac{\partial \rho(x,t)}{\partial t} = \nabla \cdot (\nabla V(x)\rho(x,t)) + \frac{\sigma^2}{2}\Delta\rho(x,t) + \sigma\nabla \cdot \left(Z(x,t)\sqrt{\rho(x,t)}\right).$$

The derivation is only valid if the number density stays a sum of Dirac delta functions for all time. Trajectories of $\{X_i(t)\}_t$ should approximately solve the SPDE when considered as empirical densities $\rho(x,t) = \sum_i \delta(x - X_i(t))$. But in the following, we instead interpret the SPDE as a justified model for the diffusion of any number densities $\rho(x,t)$. Then also $\sqrt{\rho(x,t)}$ is well-defined. We presume that this interpretation is only appropriate if we have many agents, i.e. when N is large. Therefore we are in the meso-scale modeling regime.

Remark. There are still some open questions regarding this derivation. We only know that the noise terms $\tilde{\xi}$ and ξ agree in their mean function and covariance function. Further studies should be concerned with the approximation quality of $\tilde{\xi}$ to ξ. Moreover, we would like to be able to quantify when the replacement of the density of Dirac delta functions by a agent number density becomes reasonable.

4.2.2. Including Interaction Rules for Agents

In the following we want to derive the interaction dynamics for agent densities. Recall that each agent i in the agent-based description can change its type via the N_R interaction rules. The changes are modeled as Markov jump processes

$$Y_i(t) = Y_i(0) + \sum_{r=1}^{N_R} \mathcal{P}_i^r \left(\int_0^t \lambda_i^r(t')dt' \right) v_r, \tag{4.2.9}$$

where $\mathcal{P}_i^r(t)$ denote i.i.d. unit-rate Poisson processes. The transition rate function for agent i and interaction rule R_r: $T_s + T_{s''} \to T_{s'} + T_{s''}$ is given by

$$\lambda_i^r(t) = \lambda_i^r(A(t), Y(t)) = \sum_{j=1}^{N} A_{ji}(t)\ \gamma_{\text{micro}}^r\ \mathbb{1}_{\{s''\}}(Y_j(t))\ \mathbb{1}_{\{s\}}(Y_i(t)).$$

In this ABM formulation, agents can only interact with agents that are in their neighborhood, given by the time-evolving adjacency matrix $A(t)$. A more detailed model formulation is outlined in Section 3.2.2.

In [31] an SPDE for the meso-scale description of interactions between agents is postulated, in the following we outline the derivation and thereby motivate that this SPDE is a reasonable choice.

In order to derive a meso-scopic description of the agent-based interactions, we first derive the Chemical Langevin equation (CLE) [19, 20]. The CLE serves as an SDE approximation to (4.2.9) for many agents and many interactions in a well-mixed system, i.e. in a non-spatial system. Secondly, we show that the postulated SPDE in [31] reduces to the Chemical Langevin equation cell-wise when projecting the SPDE onto a fine partition of the domain D. Thus we can view this SPDE as a spatially extended CLE that serves as an appropriate model for large interacting agent densities.

First Step. We consider the well-mixed situation of agents distributed in the volume $|D| = \int_D dx$. Well-mixedness means that agents meet many times without interacting and thus diffusion must be much faster than interactions. For well-mixed systems spatial information does not play a role. We consider agent number densities $\rho(t) = (\rho_s(t))_{s=1,\dots,N_T}$ for the different types s, the spatial argument is not needed in this scenario. The agent number density gives the number of agents of a type s per unit volume. Then we can approximate (4.2.9) by

$$\rho_s(t) = \rho_s(0) + \sum_{r=1}^{N_R} \frac{\nu_s^r}{|D|} \mathcal{P}^r \left(\int_0^t |D| a^r(\rho(t'))dt' \right), \tag{4.2.10}$$

with[5] $a^r(\rho(t)) := \gamma^r_{\text{meso}} \rho_1(t)\rho_2(t)$ and $\mathcal{P}^r(t)$ are i.i.d. unit-rate Poisson processes. Since the Poisson process gives discrete jumps in the number of agents, we have to divide by the volume $|D|$ to get a density.

In the well-mixed scenario and assuming many agents and many interactions, we are now interested in deriving an approximation to the jump process (4.2.10). Assuming an infinitesimal time interval $[t, t+\Delta t)$ such that $a^r(\rho(t))$ is approximately constant on $[t, t+\Delta t)$, we can replace $\mathcal{P}^r \left(\int_t^{t+\Delta t} |D| a^r(\rho(t'))dt' \right)$ by $\mathcal{P}^r \left(|D| a^r(\rho(t))\Delta t \right)$, such that we arrive at

$$\frac{\rho_s(t+\Delta t) - \rho_s(t)}{\Delta t} = \sum_{r=1}^{N_R} \frac{\nu_s^r}{|D|\Delta t} \mathcal{P}^r \left(|D| a^r(\rho(t))\Delta t \right). \tag{4.2.11}$$

Further by requiring $|D| a^r(\rho(t))\Delta t \gg 1$ and making use of the Central Limit Theorem, the Poisson process with rate $|D| a^r(\rho(t))\Delta t$ can be well approximated by

$$\mathcal{N}(|D| a^r(\rho(t))\Delta t, |D| a^r(\rho(t))\Delta t),$$

[5] Note that the rates γ^r_{micro} and γ^r_{meso} have different units: the micro-scale rates have units of inverse time, whereas the meso-scale rates have units of volume × inverse time.

where $\mathcal{N}(m, \sigma^2)$ is the normal distribution with mean m and variance σ^2. Using $\mathcal{N}(m, \sigma^2) = m + \sigma\,\mathcal{N}(0,1)$, Equation (4.2.11) becomes

$$\begin{aligned}\frac{\rho_s(t+\Delta t) - \rho_s(t)}{\Delta t} &\approx \sum_{r=1}^{N_R} \frac{\nu_s^r}{|D|\Delta t}\left(|D|a^r(\rho(t))\Delta t + \sqrt{|D|a^r(\rho(t))\Delta t}\;\zeta_r(t)\right)\\ &= \sum_{r=1}^{N_R} \nu_s^r\left(a^r(\rho(t)) + \sqrt{\frac{a^r(\rho(t))}{|D|\Delta t}}\zeta_r(t)\right)\end{aligned}$$

with i.i.d. $\zeta_r(t) \sim \mathcal{N}(0,1)$.

The last step is to let $\Delta t \to 0$, then $\frac{1}{\sqrt{\Delta t}}\zeta_r(t)$ can be replaced by a white noise process $Z_r(t) \sim \lim_{\Delta t \to 0} \mathcal{N}(0, 1/\Delta t)$ with the property $\mathbb{E}\,(Z_r(t)Z_{r'}(t')) = \delta_{rr'}\delta(t-t')$. And with this we can write the following SDE, also called the Chemical Langevin equation, as an approximation to (4.2.10)

$$\frac{d\rho_s(t)}{dt} = \sum_{r=1}^{N_R} \nu_s^r\left(a^r(\rho(t)) + \sqrt{\frac{a^r(\rho(t))}{|D|}}Z_r(t)\right).$$

Second Step. In our agent-based model, we do not have a completely well-mixed system as agents are only interacting with other near-by agents and diffusion is not necessarily that fast. Therefore, we are going one step further and postulate that in continuous space interactions can be modeled by the following stochastic PDE [31]

$$\frac{d\rho_s}{dt}(x,t) = \sum_{r=1}^{N_R} \nu_s^r\left(a^r(\rho(x,t)) + \sqrt{a^r(\rho(x,t))}Z_r(x,t)\right) \tag{4.2.12}$$

with space-time white noise $Z_r(x,t)$ and spatial agent densities $\rho(x,t) = (\rho_s(x,t))_{s=1,\dots,N_T}$. When projecting the solution of this equation onto the space spanned by piecewise constant functions on a fine grid, the SPDE approximately reduces to the CLE in each grid cell. Especially for the grid size approaching zero, the error in the approximation becomes negligible. And in that way the postulated SPDE model is consistent with the CLE and can be seen as a spatial extension of the usual Chemical Langevin equation.

For simplicity we restrict ourselves to the 1D case. We will compute the orthogonal projection from the SPDE onto an equidistant grid in 1D with grid size h, i.e. grid cells $[x_j, x_{j+1}) = [x_j, x_j + h)$. Projecting $\rho_s(x,t)$ onto the space spanned by orthonormalized indicator functions on the grid, we arrive at the projected density

$$\sum_j \rho_{s,j}(t)\mathbb{1}_{[x_j,x_{j+1})}(x),$$

where $\mathbb{1}_{[x_j,x_{j+1})}(x)$ is the indicator function on the j^{th} interval, i.e. mapping the $x \in [x_j, x_{j+1})$ to 1 and else to 0. The coefficients $\rho_{s,j}(t)$ are the average density in each of the cells $[x_j, x_{j+1})$

and are therefore independent of the position inside the cell, i.e.

$$\rho_{s,j}(t) = \frac{1}{h}\int_{x_j}^{x_{j+1}} \rho_s(x,t)dx.$$

Next, projecting the interaction SPDE (4.2.12), more precisely the solution of the SPDE, onto the grid, we get for each cell j

$$\frac{1}{h}\int_{x_j}^{x_{j+1}} \frac{d\rho_s}{dt}(x,t)dx = \frac{1}{h}\sum_{r=1}^{N_R}\left(\nu_s^r \int_{x_j}^{x_{j+1}} a^r(\rho(x,t))dx + \nu_s^r \int_{x_j}^{x_{j+1}} \sqrt{a^r(\rho(x,t))}Z_r(x,t)dx\right).$$

We assume that the grid cells are small or well-mixed enough, such that the density $\rho_s(x,t)$ is approximately constant on each cell, leading to

$$\frac{1}{h}\int_{x_j}^{x_{j+1}} a^r(\rho(x,t))dx \approx a^r(\rho_j(t))$$

and

$$\frac{1}{h}\int_{x_j}^{x_{j+1}} \sqrt{a^r(\rho(x,t))}Z_r(x,t)dx \approx \frac{1}{h}\sqrt{a^r(\rho_j(t))}\int_{x_j}^{x_{j+1}} Z_r(x,t)dx.$$

Since $\frac{1}{h}\int_{x_j}^{x_{j+1}} Z_r(x,t)dx$ is just $\frac{1}{\sqrt{h}}Z_{r,j}(t)$, with $Z_{r,j}(t)$ denoting i.i.d. white noise in time, we finally are back to the Chemical Langevin equation in a cell of size h,

$$\frac{d\rho_{s,j}}{dt}(t) = \sum_{r=1}^{N_R}\left(\nu_s^r a^r(\rho_j(t)) + \nu_s^r\sqrt{\frac{a^r(\rho_j(t))}{h}}Z_{r,j}(t)\right).$$

This suggests that the SPDE (4.2.12), i.e. the spatial extension of the CLE, is consistent with the usual non-spatial CLE. Projecting the SPDE onto the space spanned by indicator functions on a fine grid (such that we can assume the agent density to be constant on each grid cell), the SPDE reduces to a set of CLEs. The CLE in turn resembles the ABM when assuming many agents and fast interactions.

4.3. Discretizing the system of SPDEs

In the previous sections we derived and considered a model for the evolution of stochastic agent densities $\rho_s(x,t) : D \times [0,T] \to \mathbb{R}_{\geq 0}$. In the following we will switch to a different interpretation and rewrite the system of SPDEs (4.1.4) as a system of stochastic ODEs describing the evolution of stochastic processes $\{\rho_s(t)\}_{t\in[0,T]}$, where each $\rho_s(t)$ is a function on D.

By writing the system of SPDEs (4.1.4) as a system of SDEs on an infinite dimensional space, we can introduce the cylindrical Wiener process expansion $W(t)$ (Definition 2.2.3), whose time derivative is space-time white noise. The system of equations describing the evolution of $\rho_s(t)$

for agent types $s = 1, \dots, N_T$ then reads

$$\begin{aligned} d\rho_s(t) =& \frac{\sigma^2}{2}\Delta\rho_s(t)dt + \nabla\cdot\left(\nabla V \rho_s(t)\right)dt + \sigma\nabla\cdot\left(\sqrt{\rho_s(t)}dW_s^{\mathcal{D}}(t)\right) \\ &+ \sum_{r=1}^{N_R}\nu_s^r\left(a^r(\rho(t))dt + \sqrt{a^r(\rho(t))}dW_r^{\mathcal{I}}(t)\right) \\ \rho_s(0) =& \rho_{s,0}, \end{aligned} \tag{4.3.1}$$

with $W_s^{\mathcal{D}}(t)$ denoting a d-dimensional vector of cylindrical Wiener processes and $W_r^{\mathcal{I}}(t)$ denoting cylindrical Wiener processes for the interaction dynamics. We further denote the linear part of this evolution SPDE by

$$-\mathcal{A}\rho_s(t) := \frac{\sigma^2}{2}\Delta\rho_s(t) + \nabla\cdot\left(\nabla V\rho_s(t)\right).$$

The operator $-\mathcal{A}$ also entails the Neumann boundary conditions. Since we are modeling a closed agent system, we require there to be no agent density flux across the boundary, i.e. $\nabla\rho_s(t)\cdot\hat{u} = 0$ on δD with $\hat{u}$ denoting the unit normal vector to the boundary.

For each interaction rule $R_r : T_s + T_{s''} \to T_{s'} + T_{s''}$ with $s, s', s'' \in \{1, \dots, N_T\}$, the term

$$F_s(\rho(t)) := \sum_{r=1}^{N_r}\nu_s^r a^r(\rho(t))$$

is non-linear due to the coupling of the two interacting densities in $a^r(\rho(t)) = \gamma^r_{\text{meso}}\rho_s(t)\rho_{s''}(t)$.

This new perspective (4.3.1) enables us to design a simulation approach for sampling trajectories based on the Finite element method. The FE method has the advantage that one can treat complicated boundaries and use complex triangulations. Besides, by considering the weak form we can move the divergence operator from $\sigma\nabla\cdot\left(\sqrt{\rho_s(t)}dW_s^{\mathcal{D}}(t)\right)$ onto the test functions. This suggests the use of the FE method over e.g. Finite Differences or the Finite Volume method.

Remark. In [31] a similar SPDE is solved using the Finite-Volume (FV) method. The FV method has the nice property that it conserves the density. Further in [31], the Gaussian noise in the interaction part is after the discretization replaced by Poisson noise for each grid cell. They argue that this is a much more accurate modeling approximation.

4.3.1. Finite Element Formulation

For developing the discretization scheme, we follow the general idea of the approach introduced in Section 2.2.3 due to the similarities of (4.3.1) to the semilinear evolution SPDE (2.2.6) introduced therein. Since it is still an open question, whether solutions to (4.3.1) exist, we will in the following assume that we can still discretize and find discretized solutions. For the evolution SPDE (2.2.6) it can be proved that a weak form exists under certain requirements on $\mathcal{A}$, the non-linearity F and the noise terms. In the case of (4.3.1), these are probably not fulfilled due

to the irregular noise terms. Therefore the theoretical backbone of the following discretization should be investigated further.

In order to formulate the Galerkin approximation, we first have to derive the weak form by multiplying Equation (4.3.1), which is interpreted as an integral equation, by test functions w and integrating over the domain D.

The weak formulation of (4.3.1) consists of finding $\rho_s(t)$ for all agent types $s = 1, \dots, N_T$ such that[6]

$$\begin{aligned}\langle \rho_s(t), w\rangle =& \langle \rho_{s,0}, w\rangle + \int_0^t \left(-\frac{\sigma^2}{2}\langle \nabla\rho_s(t'), \nabla w\rangle + \langle \nabla\cdot\left(\nabla V \rho_s(t')\right), w\rangle + \sum_{r=1}^{N_R} \nu_s^r \langle a^r(\rho(t')), w\rangle \right) dt' \\ &- \sigma \int_0^t \langle \sqrt{\rho_s(t')} dW_s^{\mathcal{D}}(t'), \nabla w\rangle + \sigma \int_0^t \left(\int_{\delta D} (\sqrt{\rho_s(t')} dW_s^{\mathcal{D}}(t')\cdot \hat{u}) w \; dx \right) \\ &+ \sum_{r=1}^{N_R} \int_0^t \nu_s^r \langle \sqrt{a^r(\rho(t'))} dW_r^{\mathcal{I}}(t'), w\rangle \quad \forall w, \; \forall t \in [0,T]. \end{aligned} \tag{4.3.2}$$

To get to Equation (4.3.2), we made use of partial integration to shift some regularity requirements from $\rho_s(t)$ onto the test functions w. In particular,

$$\langle \Delta\rho_s(t), w\rangle = \int_D \Delta\rho_s(t) w \; dx = -\int_D \nabla\rho_s(t)\cdot\nabla w \; dx + \int_{\delta D} (\nabla\rho_s(t)\cdot\hat{u}) w \; dx = -\langle \nabla\rho_s(t), \nabla w \; \rangle$$

making use of the no-flux boundary conditions $\nabla\rho_s(t)\cdot\hat{u} = 0$ on δD. Further, we want to shift the divergence operator from the space-time white noise onto the test functions. Using partial integration again we have

$$\langle \nabla\cdot(\sqrt{\rho_s(t)} dW_s^{\mathcal{D}}(t)), w\rangle = -\langle \sqrt{\rho_s(t)} dW_s^{\mathcal{D}}(t), \nabla w\rangle + \int_{\delta D} (\sqrt{\rho_s(t)} dW_s^{\mathcal{D}}(t)\cdot\hat{u}) w \; dx.$$

Letting $\tilde{V}$ denote the finite dimensional solution space and test function space spanned by the basis $\{\phi_i\}_{i=0}^n$, the Galerkin approximation then consists of finding $\tilde{\rho}_s(t) \in \tilde{V}$ for each agent type $s = 1, \dots, N_T$ such that

$$\begin{aligned}\langle \tilde{\rho}_s(t), \phi_i\rangle =& \langle \rho_{s,0}, \phi_i\rangle + \int_0^t \left(-\frac{\sigma^2}{2}\langle \nabla\tilde{\rho}_s(t'), \nabla\phi_i\rangle + \langle \nabla\cdot\left(\nabla V \tilde{\rho}_s(t')\right), \phi_i\rangle + \sum_{r=1}^{N_R} \nu_s^r \langle a^r(\tilde{\rho}(t')), \phi_i\rangle \right) dt' \\ &- \sigma \int_0^t \langle \sqrt{\tilde{\rho}_s(t')} dW_s^{\mathcal{D}}(t'), \nabla\phi_i\rangle + \sigma \int_0^t \left(\int_{\delta D} (\sqrt{\tilde{\rho}_s(t')} dW_s^{\mathcal{D}}(t')\cdot \hat{u}) \phi_i \; dx \right) \\ &+ \sum_{r=1}^{N_R} \int_0^t \nu_s^r \langle \sqrt{a^r(\tilde{\rho}(t'))} dW_r^{\mathcal{I}}(t'), \phi_i\rangle \quad \forall i, \; \forall t \in [0,T]. \end{aligned} \tag{4.3.3}$$

[6]The inner product is given by $\langle u, v\rangle = \int_D uv \; dx$.

We can expand the solution as a linear combination of the basis functions $\{\phi_j\}_{j=0}^n$ with time-dependent coefficients $\beta_{s,j}(t)$,

$$\tilde{\rho}_s(t) = \sum_{j=0}^{n} \beta_{s,j}(t)\phi_j.$$

Then following the same steps as for semilinear evolution SPDEs in Section 2.2.3 and by defining matrices

$$\begin{aligned} C_{ji} &= \langle \phi_j, \phi_i \rangle \\ A_{ji} &= \frac{\sigma^2}{2}\langle \nabla\phi_j, \nabla\phi_i \rangle - \langle \nabla \cdot (\nabla V \phi_j), \phi_i \rangle \end{aligned}$$

for $i, j = 0, \ldots, n$, and coefficient vector $\beta_s(t) = (\beta_{s,j}(t))_{j=0}^n$, we can write (4.3.3) as

$$\begin{aligned} \sum_{j=0}^{n} d\beta_{s,j}(t)C_{ji} = &- \sum_{j=0}^{n} \beta_{s,j}(t)A_{ji}\ dt + \langle F_s\left(\tilde{\rho}(t')\right), \phi_i \rangle\ dt - \sigma\langle \sqrt{\tilde{\rho}_s(t)} dW_s^{\mathcal{D}}(t), \nabla\phi_i \rangle \\ &+ \sigma \int_{\delta D} (\sqrt{\tilde{\rho}_s(t)} dW_s^{\mathcal{D}}(t) \cdot \hat{u})\phi_i\ dx + \sum_{r=1}^{N_R} \nu_s^r \langle \sqrt{a^r(\tilde{\rho}(t))} dW_r^{\mathcal{I}}(t), \phi_i \rangle, \quad \forall i = 0, \ldots, n. \end{aligned}$$

We are also denoting the vector of the non-linear term by

$$F_s(t) = \langle F_s(\tilde{\rho}(t)), \phi_i \rangle_{i=0}^n.$$

By truncating the noise to M dimensions (recall the Definition 2.2.3 and the noise truncation in (2.2.10)) and by defining

$$dW^M(t) = (dB_m(t))_{m=1}^M,$$

$$G_s^{\mathcal{D}}(t)_{im} = -\sigma\langle \sqrt{\tilde{\rho}_s(t)}\chi_m, \nabla\phi_i \rangle + \sigma \int_{\delta D} (\sqrt{\tilde{\rho}_s(t)}\chi_m \cdot \hat{u})\phi_i\ dx,$$

and

$$G_{s,r}^{\mathcal{I}}(t)_{im} = \nu_s^r \langle \sqrt{a^r(\tilde{\rho}(t))}\chi_m, \phi_i \rangle,$$

our Galerkin approximation finally reads

$$C d\beta_s(t) = (-A\beta_s(t) + F_s(t))\ dt + G_s^{\mathcal{D}}(t)\ dW_s^{\mathcal{D},M}(t) + \sum_{r=1}^{N_R} G_{s,r}^{\mathcal{I}}(t)\ dW_r^{\mathcal{I},M}(t). \tag{4.3.4}$$

The last step is to discretize in time. Denoting functions at time $t_k = k\Delta t$ by a subscript k, e.g. $\beta_s(t_k) = \beta_{s,k}$, the Euler-Maruyama time-discretization of (4.3.4) is the recursion for $k = 0, \ldots, K-1$

$$\beta_{s,k+1} = (C + A\Delta t)^{-1} \left(C\beta_{s,k} + F_{s,k}\Delta t + G_{s,k}^{\mathcal{D}} \Delta W_{s,k}^{\mathcal{D},M} + \sum_{r=1}^{N_R} G_{s,r,k}^{\mathcal{I}} \Delta W_{r,k}^{\mathcal{I},M} \right). \tag{4.3.5}$$

The Brownian increments

$$\Delta W_k^M = (\sqrt{\Delta t}\zeta_{m,k})_{m=1}^M$$

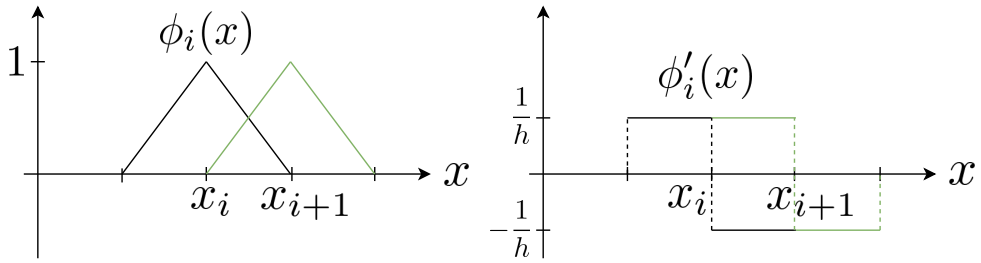


Figure 4.2.: Hat functions ϕ_i and their derivative ϕ_i' in one dimension.

have to be sampled for each time step by drawing i.i.d. $\zeta_{m,k} \sim \mathcal{N}(0,1)$.

Remark. One numerical problem is that the agent density can become negative in the simulations. This is only due to the discretization. When modeling the interactions between different densities and for a not too small time step Δt, it can happen that too much density is subtracted from one type. One possibility for tackling this problem in the implementation is to work with $\max(0, \beta_{s,j,k})$ instead of $\beta_{s,j,k}$ and thereby ensure its non-negativity [31].

4.3.2. Assembling Matrices in 1D

Sampling trajectories of the SPDE can be done by iteratively solving (4.3.5) in parallel for each agent type. We will explain in the following how to assemble the necessary matrices for a one-dimensional domain. The domain D is partitioned into an equidistant grid supporting the hat functions that are spanning the finite-dimensional space $\tilde{V}$.

Construction of the Finite Element Space

For the finite-dimensional subspace $\tilde{V}$ of dimension $n+1$ we will consider the space spanned by hat functions $\{\phi_i\}_{i=0}^n$ (i.e. piecewise linear functions) on an equidistant grid. This will simplify matrix computations, since the resulting matrices will be sparse due to the hat functions being mostly zero throughout D. For $D = [0, a]$, we take an equidistant grid with step size $h = \frac{a}{n}$ such that we have $x_0 = 0 < x_1 = h < x_2 = 2h < \cdots < x_n = a$ as our triangulation.

The i^{th} hat function is defined as

$$\phi_i(x) = \begin{cases} 0 & x < x_{i-1} \\ \frac{x - x_{i-1}}{h} & x_{i-1} \le x < x_i \\ 1 - \frac{x - x_i}{h} & x_i \le x < x_{i+1} \\ 0 & x \ge x_{i+1}. \end{cases}$$

These have the nice properties $\phi_i(x_j) = \delta_{ij}$ and $\phi_i(x)\phi_j(x) = 0$ for all $|i - j| > 1$.

The derivative of the hat function is piecewise constant on the grid and given by

$$\phi_i'(x) = \begin{cases} 0 & x < x_{i-1} \\ \frac{1}{h} & x_{i-1} \le x < x_i \\ -\frac{1}{h} & x_i \le x < x_{i+1} \\ 0 & x \ge x_{i+1}. \end{cases}$$

The hat functions and their derivatives are shown in Figure 4.2.

Some matrices of (4.3.5) can be computed analytically from the definition of the hat functions. One finds that

$$\langle \phi_j, \phi_i \rangle = \begin{cases} \frac{2h}{3} & j = i \neq 0, n \\ \frac{h}{6} & j = i+1 \text{ or } j = i-1 \\ \frac{h}{3} & i = j = 0 \text{ or } i = j = n \\ 0 & \text{else} \end{cases}$$

$$\langle \phi_j', \phi_i' \rangle = \begin{cases} \frac{2}{h} & j = i \neq 0, n \\ -\frac{1}{h} & j = i+1 \text{ or } j = i-1 \\ \frac{1}{h} & i = j = 0 \text{ or } i = j = n \\ 0 & \text{else} \end{cases}$$

for $i, j = 0, \dots, n$.

Numerical Assembly

To assemble A_{ji}, we can use $\langle \phi_j', \phi_i' \rangle$ and additionally integrate

$$\langle \nabla \cdot (\nabla V \phi_j), \phi_i \rangle = \langle \Delta V \phi_j, \phi_i \rangle + \langle \nabla V \cdot \nabla \phi_j, \phi_i \rangle = \int_D (\Delta V \phi_j \phi_i + \nabla V \cdot \nabla \phi_j \phi_i) \, dx$$

numerically using the trapezoidal rule.

We can approximate the integral of a function $u : [a, b] \to \mathbb{R}$ over the interval $[a, b]$ with the trapezoidal rule by approximating the area under the graph by a trapezoid

$$\int_a^b u(x)dx \approx \frac{1}{2(b-a)} \left(u(a) + u(b) \right).$$

By additionally subdividing $[a, b]$ into grid cells and applying the trapezoidal rule on every subinterval, we can improve the approximation.

We further assemble the non-linear and the noise matrices on the basis of the trapezoidal rule. As a basis for the expansion of the cylindrical Wiener process we choose $\chi_m(x) = \sqrt{\frac{2}{a}} \sin\left(\frac{\pi m x}{a}\right)$, $m \in \mathbb{N}$, similar as in the space-time white noise expansion (2.2.4).

Remark. For equidistant grids, we can assemble matrices as shown above. But the Finite Element method can be used to treat much more complex domains and triangulations. This is where one of its strengths lies. One can then assemble the matrix elements by defining a reference grid cell and transforming from any grid cell to the reference grid cell in order to do the computations on the reference cell before transforming back [29].

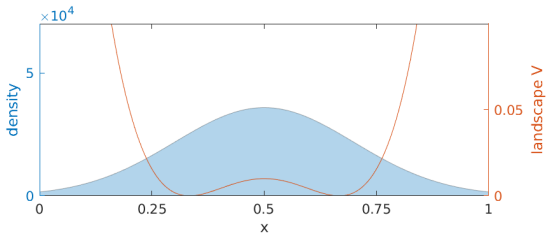
4.4. Numerical Example: Innovation Spreading in a Double Well Landscape

We again consider the model example as introduced in Sections 3.1 and 3.4 of agents diffusing in a double well landscape and the spreading of an innovation among agents. However this time our model is formulated as a system of SPDEs. Since we assumed large agent populations for the approximation of the ABM by the SPDE model, we will consider a system of $N = 20000$ agents on $D = [0, 1]$. The simulation of trajectories of the density-based model is based on the discretization as introduced in Section 4.3.

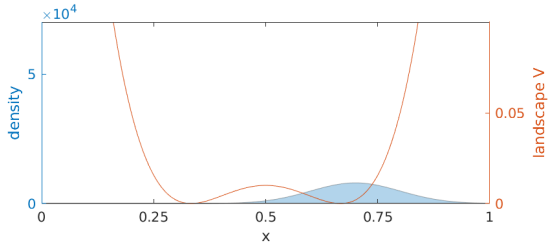
In Figure 4.3 we plot the results of one simulation showing the evolution of the number density of non-adopters $\rho_1(x, t)$ and the density of adopters $\rho_2(x, t)$ for $t \in [0, 2]$. The emergent dynamics are very similar to the ABM dynamics in Section 3.4. Again the agent density is clustered around the two wells. The spreading of the innovation inside the wells is fast, but there is a long time gap for the innovation to spread from one well to another. However the global dynamics agree only qualitatively not quantitatively between the ABM and the SPDE model because of the differently chosen parameters.
We observe that the stochasticity inherent in the model is still visible on the global scale, the agent densities are seemingly noisy. But compared to the ABM example (Section 3.4), the noisiness of the densities has reduced. The probable reason is that we consider a system with a much larger agent population and that their behaviour averages out on a larger scale. However another explanation could be that the approximation does not represent the randomness well. In the next chapter we will study the approximation quality of the reduced model to the agent-based dynamics in more detail.

Further, we are studying an ensemble of trajectories solving the SPDE by computing some observable of the dynamics. We consider the evolution of the number of T_2 agents in time for an ensemble of 50 realizations, see Figure 4.4. Similar to the ABM example, there are some trajectories deviating from the ensemble mean, but only slightly.

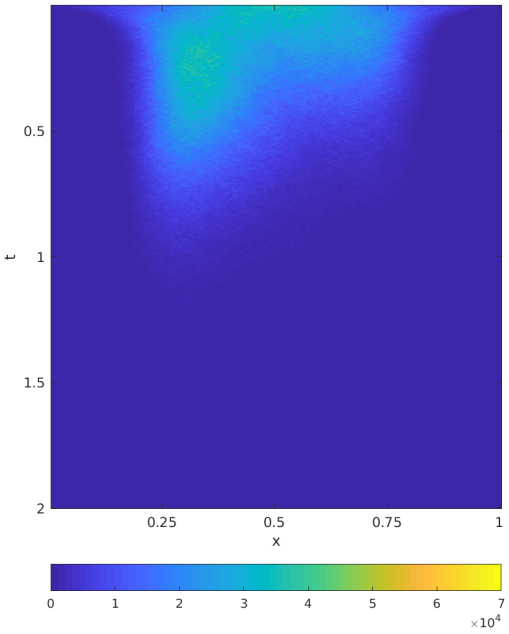
In the following chapter we will study and compare the ABM and its approximation from a numerical perspective. The goal is to quantify, at least for our toy example, the approximation quality of the density-based model to the ABM and the computational effort of both models.



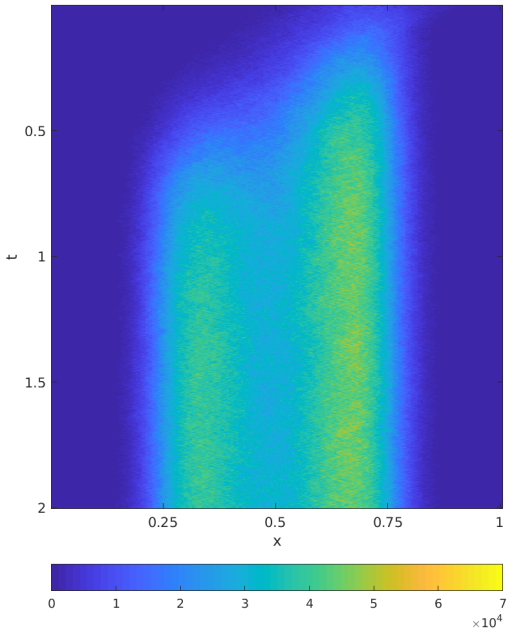
(a) Initial density of non-adopters at time $t = 0$.



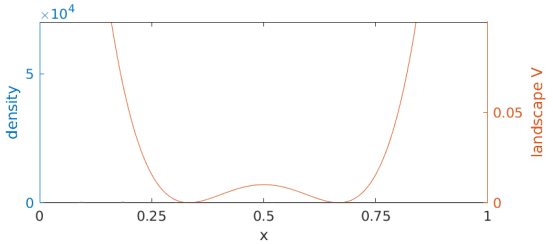
(b) Initial density of adopters at time $t = 0$.



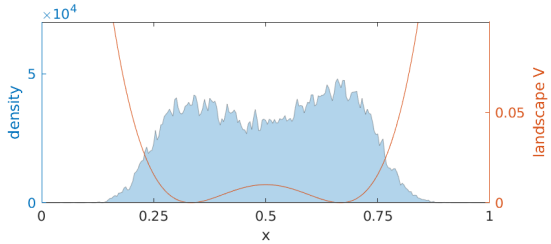
(c) Evolution of the non-adopter density $\rho_1(x,t)$.



(d) Evolution of the density of adopters $\rho_2(x,t)$.



(e) Final density of non-adopters at time $t = 2$.



(f) Final density of adopters at time $t = 2$.

Figure 4.3.: Agents diffusing in a double well and innovation spreading, initially 2000 agents have adopted the innovation, 18000 agents are non-adopters. The initial density of non-adopters is a Gaussian function centered at $x = 0.5$ with standard deviation 0.1, whereas the density of adopters is initially a Gaussian placed at $x = 0.7$ with standard deviation 0.1. Further parameters: $\sigma = 0.25$, $\gamma^1_{\text{meso}} = 0.0002$, $\Delta t = 0.0001$, $h = 0.004$, $M = 256$.

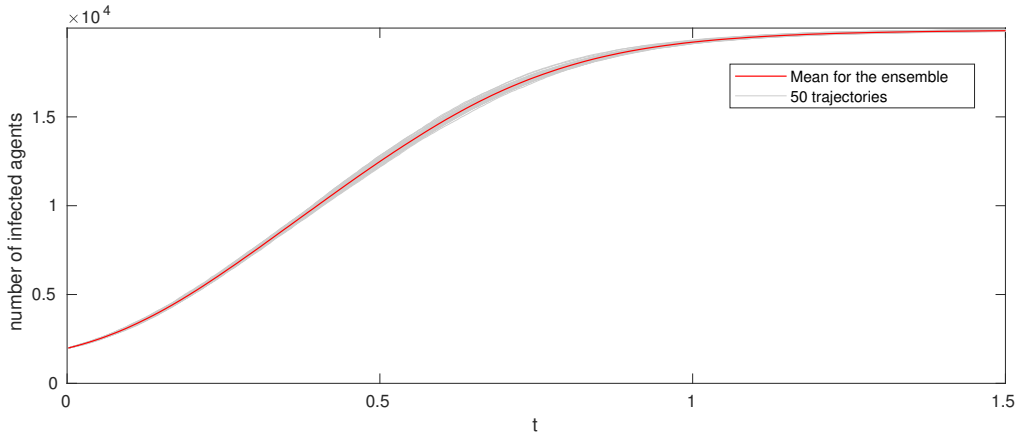


Figure 4.4.: For an ensemble of 50 simulations we study an observable of the system, namely the evolution of the number of adopters in time. At time t the number of agents of type T_2 is given by $\int_D \rho_2(x,t)\ dx$.

5. Comparison of the Agent-based and Density-based Model

Many models of real-world dynamics pose challenges regarding their simulation due to a complex model formulation. We consider the agent-based model as introduced in Chapter 3 as the ground-truth model for a system of diffusing and interacting agents. However due to its high model complexity, simulations are often too expensive. Moreover, the stochasticity in the model asks for repeated simulations. Reduced model descriptions are therefore needed, these reduced models should have a small approximation error as well as a being computationally much more efficient.

Depending on the spatial and the population scale of the underlying process of interest, different approximations to the agent-based model are possible and reasonable.

On the spatial scale we can distinguish between:

- models with fully continuous space, e.g. [31].
- models with a spatial partition into non-overlapping well-mixed cells, the partitioning can be based on natural boundaries or just a regular grid, e.g. [28, 48, 6].
- well-mixed model descriptions, where spatial information is neglected, e.g. [19, 20].

For different population scales, i.e. different agent numbers, we categorize into models of:

- small populations that are described in terms of discrete agents evolving stochastically, e.g. [20].
- large agent numbers described as densities and whose changes are modeled continuously but still including randomness, e.g. [19, 20, 31].
- very large, nearly infinite populations, modeling density evolutions deterministically, e.g. [20, 41].

However in this thesis we are only concerned with systems including full spatial resolution. The agent-based model is valid on all population scales but we expect its computational feasibility only on the smallest population scale. By replacing discrete agents by agent densities and by assuming a large number of agents, we derived a reduced SPDE model in Chapter 4. The SPDE description is supposedly a good approximation to the ABM for large agent numbers, whilst still including stochasticity.

In this chapter we want to computationally confirm this perspective and make it more concrete by studying our guiding example for different populations scales using the ABM and the reduced density-based model. We are setting up a computational experiment to study the numerical effort per time step of both approaches and to investigate how well the density-based model approximates the ABM for increasing agent populations.

5.1. Experiments on Computational Effort and Approximation Quality

In this section we are computationally studying how the agent-based and density-based model compare in computation time and finding out at which agent numbers N the density-based model starts to be a good approximation to the ABM. We will compare the two models using our guiding example of innovation spreading as introduced in Sections 3.1 and 3.4.

Before setting up the experiments, we first we have to ensure the consistency between the parameters for both models. The diffusion parameters are the same for the ABM and the SPDE description, but the interaction parameters have to be converted. In the agent-based model, pairs of agents are interacting at a fixed rate γ^r_{micro} (units of inverse time) for each interaction rule R_r whenever they are within a radius d_{int} of each other. In the SPDE formulation on the other hand, agent densities are interacting point-wise with constant rate factor γ^r_{meso} (units of volume × inverse time). The parameter conversion is still very much an open problem. We will below explain two strategies [14, 15].

Strategy 1 (Data-driven). For many applications, we can carry out experiments and then estimate the interaction rates γ^r_{meso} from the measured data. To convert γ^r_{meso} to the ABM parameters, we set the radius d_{int} to be the sum of the radii of the two interacting agents (e.g. molecular radius, human radius of infection) and then determine γ^r_{micro} such that it fits best to the outcome of the experiments.

Strategy 2. For systems where the diffusion of agents is fast enough such that agents are locally uniformly mixed and there are no spatial correlations between different agent types, we can simply convert parameters from the micro-scale to the meso-scale by $\gamma^r_{\text{meso}} = \gamma^r_{\text{micro}} V_{\text{int}}$ [15]. In that way interactions take place within the interactive volume V_{int}, the volume of a ball of radius d_{int}.

Given these conversion strategies, we are ready to set up a numerical experiment to compare the computational effort and the approximation quality of the density-based model to the ABM for increasing agent numbers N.

Experimental Set-up. We consider again our guiding example of agents spreading an innovation according to the rule $R_1 : \ T_1 + T_2 \to 2\ T_2$, but this time agents are diffusing in the domain $D = [0, 1]$ with drift in a single well landscape $V(x) = 5(x - 0.5)^2$ centered at $x = 0.5$. As an initial configuration, 10% of agents are of type T_2 and their positions are distributed normally with mean 0.5 and standard deviation 0.02. The remaining agents are at time $t = 0$ of type T_1 and placed according to a normal distribution with mean 0.5 and standard deviation 0.15. Further parameters are $d_{\text{int}} = 0.001$, $\gamma^1_{\text{micro}} = 0.5$, $\sigma = 0.25$, $h = 0.004$, $M = 256$.
Trajectories of the ABM and density-based model are then sampled following the simulation approaches introduced in Sections 3.3 and 4.3, and using Strategy 2 to convert the interaction parameters from the micro-scale to the meso-scale.

For each population size $N \in \{50, 110, 150, 220, 330, 440, 550, 1100, 2200, 3300\}$, we need to simulate repeatedly ($\texttt{sim} = 100$) in order to compute meaningful ensemble averages. We have implemented the simulation schemes in Matlab and run our code on a computer with an Opteron 8384 CPU.

The Computational Effort. For both models and varying N, we fix the step size in time $\Delta t = 0.005$ and measure the time it takes to simulate one time step. Since both simulation approaches make use of an Euler-Maruyama time discretization, comparing the effort per time step is sensible. But space is treated differently. In the ABM, we use an Euler-Maruyama discretization for the position dynamics of each agent. Whereas in the SPDE approach, we use the scheme for each finite element. Moreover for the simulation of the ABM we have to compute pair-wise distances between agents, which scales badly for increasing agent numbers N. Thus we expect the computational effort for the ABM to depend on the number of agents, whereas for the SPDE model it should be independent of N and thus constant for increasing N.

Approximation Quality. We compute observables of the simulated dynamics for both models and increasing agent numbers. Based on these measured observables we can compare how well they agree and deduce the approximation quality of the density-based model to the ABM[1]. Possible observables are e.g. the time it takes until the agent system has reached a certain state for the first time or the state of the system at a fixed time point. Here, we study our guiding example of innovation spreading in a single well landscape. The observables we consider are (i) the time until 90% of agents are of type T_2, and (ii) the spatial distribution of type T_2 agents at a fixed model time point $t = 0.2$. We expect that with increasing N, the observables of the density-based model agree better and better with the observables of the agent-based model.

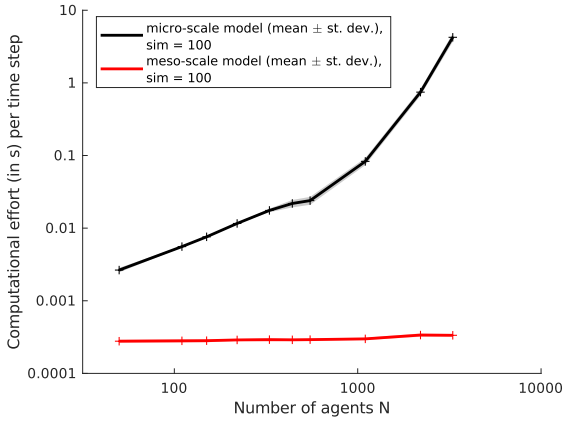
5.2. Experimental Results

In the following we come to a discussion of the results. The results of our experiments on the computational effort of both models and the approximation quality are shown in Figure 5.1.

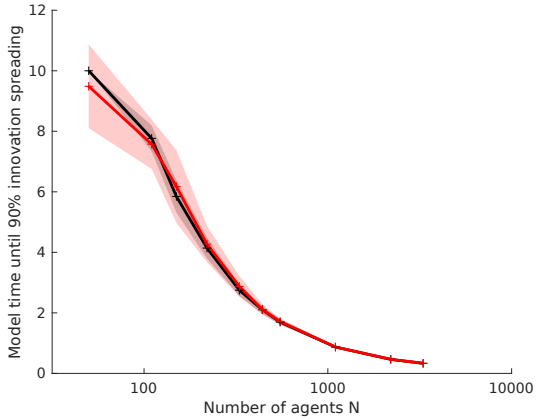
The simulation results in Figure 5.1a show that the computational effort per time step in the ABM scales exponentially, whereas the effort for the simulation of the density-based model is independent of the number of agents and several magnitudes below.

Let us now discuss the approximation quality between the models. We compare the observables of the density-based model with the observables of the ABM for increasing N as depicted in Figures 5.1b and 5.1c.
For systems of only very few agents, the observables agree only roughly in mean between the two models, but the standard deviation is much larger for the density-based model. The most likely reason is that we assumed many agents and many interactions for the derivation of the SPDE model from the ABM. Considering the agent density at a fixed time point for $N = 50$ agents in

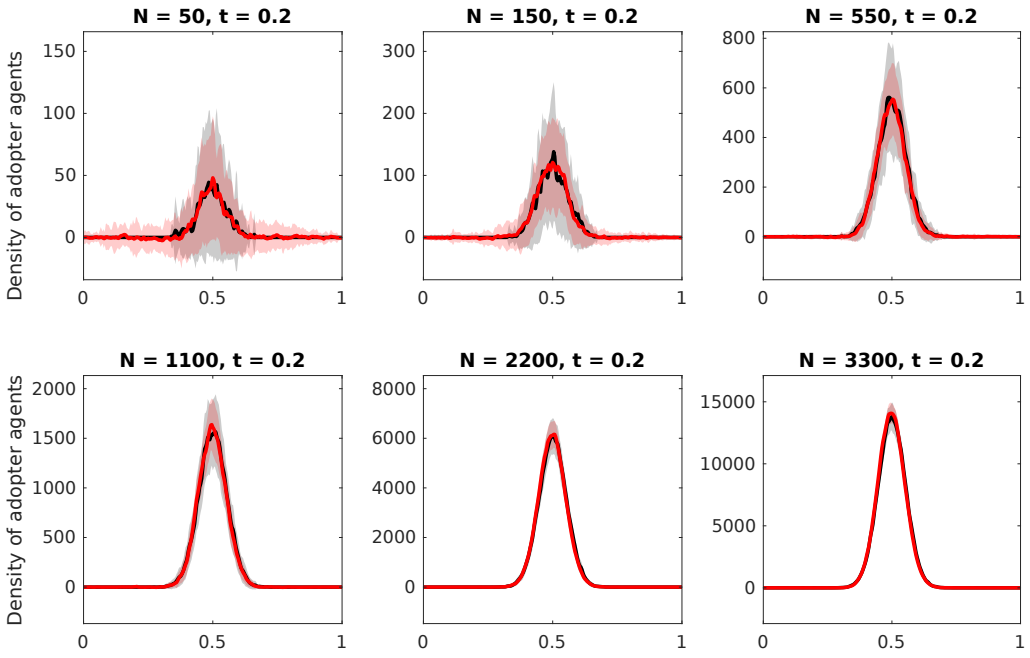
[1] Computing these observables we also make some numerical and statistical error.



(a) Computational effort per time step.



(b) Approximation quality for an observable.



(c) Approximation quality for the space-dependent observable on $D = [0, 1]$. We plot the agent density of type T_2 at time $t = 0.2$ and for both models. The curve in black gives the mean agent density for the ABM whereas the graph in red indicates the mean agent density for the SPDE model, the standard deviation is indicated by the shaded area.

Figure 5.1.: Study of the numerical effort and model approximation quality for systems of an increasing number of agents N and 100 simulations. We are again considering our guiding example of agents diffusing in a single-well and innovation spreading.

Figure 5.1c, we note that the standard deviation for the observable of the density-based model is much larger away from the center of the well, where the agent density is very low. The random terms in the SPDE dominate in the low-agent number regime and thus away from the center of the well. We can conclude that the approximation of the density-based model to the ABM for systems of few agents, in this case $N < 500$, is not very good.
For systems with a large number of agents, the observables agree well in mean and standard deviation, thus the approximation error is very low and the density-based model serves as a good approximation to the ABM in this regime.
Already for systems of $N > 3000$, the observables get close to deterministic with a very small standard deviation. Whilst the system is still slightly random on this population scale, a PDE model could in some modeling cases be sufficient to describe the system, especially for even larger numbers of agents.

Extrapolating from our experiments on a toy example, we can in conclusion say that for small agent numbers the ABM is still the best description even though it is more expensive to simulate than the SPDE model. For a larger number of agents, it is appropriate to model with SPDEs, i.e. on the meso-scale. Observables are well approximated in mean and standard deviation and the simulation approach enables many thousand simulations where before only a couple of simulations were computationally feasible. For an even larger number of agents, one can possibly approximate the SPDE by a PDE and numerically compute solutions of the PDE. Numerically solving the PDE does not required repeated simulations, since the solutions are deterministic.
In general, to draw conclusions on the minimum number of agents needed such that the replacement of the agent-based model by the reduced density-based model is adequate, one needs to consider the agent density locally and justify that the assumptions for deriving the SPDE system (Section 4.2) are fulfilled locally.

6. Conclusion and Future Outlook

In this thesis we have introduced a general agent-based model that is formulated in terms of coupled diffusion and Markov jump processes for each agent. Simulating agent-based models for real-world dynamics quickly becomes costly due to an explosion in the computational complexity for increasing agent numbers and the need for repeated simulations due to its stochastic description. For instance, when modeling the spreading of innovations in ancient times such as in [7, 8], many Monte Carlo simulation are required to capture the full spectrum of the diverse dynamics. But this becomes computationally very expensive. A thorough sensitivity analysis of the parameters demands many simulations for each parameter set and is as such not tractable.

We therefore derived, based on [31, 10], an approximation to the ABM for systems of many agents. The reduced model is given by a system of coupled stochastic PDEs propagating agent densities for the different agent types. For both models, the ABM and the reduced density-based model, we constructed and explained simulation schemes. In the last chapter, we compared the simulation effort and studied the approximation quality of the reduced density-based model to the ABM computationally on a toy example. From these computational experiments we can conclude the following. For systems of few agents, the ABM is not too costly. We consider the ABM as the ground-truth model and thus as the most accurate description of the agent system. The dynamics have to be described in terms of individual agents, since there are only very few. But for systems of many agents, we can instead use the approximation by the SPDE to study the agent system dynamics much more efficiently. In the case of our toy example this approximation is accurate in mean and standard deviation already for systems of a thousand agents. Thus this reduced model is a very promising tool for modeling and especially simulating real-world systems of large populations.

The goal of this thesis was to take a first step into the direction of modeling agent systems on the meso-scale by means of stochastic PDEs. There are however several questions that still remain unanswered and should be investigated in the future. First, model descriptions for more complicated dynamics such as including more complex interaction rules, including space or type dependent parameters or containing feedback loops, have to be derived both on the micro-scale as well as for the reduced SPDE model. Adding a feedback loop to our ABM would mean that the position dynamics affect the interaction rules and vice versa. This strong coupling will probably pose new challenges. Second, we need to better understand the approximation error of the density-based model to the agent-based model, quantitative statements about the approximation quality are still missing. Third, the existence, uniqueness and regularity of solutions to the SPDE still need to be discussed [16]. Fourth, it would be interesting to analyze the properties of the SPDE and the discretized SPDE, such as conservation properties as well as properties of the operators of the continuous SPDE system and matrices in the discretized case. Fifth, a thorough numerical analysis and comparison of different discretization strategies (e.g. Finite Element, Finite Volume method as well as temporal integrators) is still missing. Sixth, more studies on the applicability of the SPDE model to real-world dynamics are needed, such as applying the model to the spreading of the wooly sheep [7, 8].

A. Appendix

Equivalence of Covariances for the Noise Terms ξ and $\tilde{\xi}$

The covariances for ξ and $\tilde{\xi}$ (equation (4.2.6) and (4.2.7) respectively) are the same when interpreted as distributions, i.e. for all test functions $f, g \in C_0^\infty(D)$

$$\int_D \int_D \nabla_x \cdot \nabla_y (\delta(x-y)\rho(x,t)) f(x) g(y)\ dx\, dy = \int_D \int_D \Big(-\sqrt{\rho(x,t)\rho(y,t)} \nabla_x \cdot \nabla_y \delta(x-y)$$
$$+ \nabla_x \delta(x-y) \cdot \sqrt{\rho(x,t)} \nabla_y \sqrt{\rho(y,t)} + \nabla_y \delta(x-y) \cdot \sqrt{\rho(y,t)} \nabla_x \sqrt{\rho(x,t)}$$
$$+ \delta(x-y) \left(\nabla_x \sqrt{\rho(x,t)} \right)^2 \Big) f(x) g(y)\ dx\, dy.$$

Proof. We can reformulate the left hand side (LHS) as

$$\begin{aligned}
\text{LHS} &= \int_D \int_D \nabla_x \cdot \nabla_y (\delta(x-y)\rho(x,t)) f(x) g(y) dx\, dy \\
&= -\int_D \int_D \nabla_y (\delta(x-y)\rho(x,t)) \cdot \nabla_x (f(x)) g(y) dx\, dy \\
&= \int_D \int_D \delta(x-y)\rho(x,t) \nabla_x (f(x)) \cdot \nabla_y (g(y)) dx\, dy \\
&= \int_D \rho(y,t) \nabla_y (f(y)) \cdot \nabla_y (g(y)) dy,
\end{aligned}$$

using integration by parts twice and properties of the test functions and Dirac delta distributions. The right hand side (RHS) can be treated similarly, we can do the following reformulations

$$\begin{aligned}
\text{RHS} &= \int_D \int_D \Big(-\sqrt{\rho(x,t)\rho(y,t)} \nabla_x \cdot \nabla_y \delta(x-y) + \nabla_x \delta(x-y) \cdot \sqrt{\rho(x,t)} \nabla_y \sqrt{\rho(y,t)} \\
&\quad + \nabla_y \delta(x-y) \cdot \sqrt{\rho(y,t)} \nabla_x \sqrt{\rho(x,t)} + \delta(x-y) \left(\nabla_x \sqrt{\rho(x,t)} \right)^2 \Big) f(x) g(y) dx\, dy \\
&= \int_D \int_D \Big(\nabla_x (\sqrt{\rho(x,t)} f(x)) \cdot \nabla_y (\sqrt{\rho(y,t)} g(y)) - \nabla_x (\sqrt{\rho(x,t)} f(x)) g(y) \cdot \nabla_y \sqrt{\rho(y,t)} \\
&\quad - \nabla_y (\sqrt{\rho(y,t)} g(y)) f(x) \cdot \nabla_x \sqrt{\rho(x,t)} + \left(\nabla_x \sqrt{\rho(x,t)} \right)^2 f(x) g(y) \Big) \delta(x-y) dx\, dy \\
&= \int_D \Big(\nabla_y (\sqrt{\rho(y,t)} f(y)) \cdot \nabla_y (\sqrt{\rho(y,t)} g(y)) - \nabla_y (\sqrt{\rho(y,t)} f(y)) g(y) \cdot \nabla_y \sqrt{\rho(y,t)} \\
&\quad - \nabla_y (\sqrt{\rho(y,t)} g(y)) f(y) \cdot \nabla_y \sqrt{\rho(y,t)} + \left(\nabla_y \sqrt{\rho(y,t)} \right)^2 f(y) g(y) \Big) dy \\
&= \int \rho(y,t) \nabla_y (f(y)) \cdot \nabla_y (g(y)) dy.
\end{aligned}$$

Thus the LHS and RHS agree and the covariances are the same. □

Bibliography

[1] Petter Abrahamsen. A review of gaussian random fields and correlation functions.

[2] Cornelia Becker, Norbert Benecke, Ana Grabundžija, Hans-Christian Küchelmann, Susan Pollock, Wolfram Schier, Chiara Schoch, Ingo Schrakamp, Britta Schütt, and Martin Schumacher. The textile revolution. research into the origin and spread of wool production between the Near East and Central Europe. In *Space and Knowledge. Topoi Research Group Articles*, volume 6, pages 102–151, 2016.

[3] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

[4] Amit Kumar Bhattacharjee, Kaushik Balakrishnan, Alejandro L Garcia, John B Bell, and Aleksandar Donev. Fluctuating hydrodynamics of multi-species reactive mixtures. *The Journal of chemical physics*, 142(22):224107, 2015.

[5] Dirk Brockmann and Dirk Helbing. The hidden geometry of complex, network-driven contagion phenomena. *Science*, 342(6164):1337–1342, 2013.

[6] Vittoria Colizza, Romualdo Pastor-Satorras, and Alessandro Vespignani. Reaction–diffusion processes and metapopulation models in heterogeneous networks. *Nature Physics*, 3(4):276–282, 2007.

[7] Nataša Conrad, Luzie Helfmann, Johannes Zonker, Stefanie Winkelmann, and Christof Schütte. Human mobility and innovation spreading in ancient times: A stochastic agent-based simulation approach. *EPJ Data Science*, 7(1):24, 2018.

[8] Nataša Djurdjevac Conrad, Daniel Furstenau, Ana Grabundžija, Luzie Helfmann, Martin Park, Wolfram Schier, Brigitta Schütt, Christof Schütte, Marcus Weber, Niklas Wulkow, and Johannes Zonker. Mathematical modeling of the spreading of innovations in the ancient world. *eTopoi. Journal for Ancient Studies*, 7, 2018.

[9] Giuseppe Da Prato and Jerzy Zabczyk. *Stochastic equations in infinite dimensions*. Cambridge university press, 2014.

[10] David S Dean. Langevin equation for the density of a system of interacting langevin processes. *Journal of Physics A: Mathematical and General*, 29(24):L613, 1996.

[11] Steven Delong, Boyce E Griffith, Eric Vanden-Eijnden, and Aleksandar Donev. Temporal integrators for fluctuating hydrodynamics. *Physical Review E*, 87(3):033302, 2013.

[12] Masao Doi. Stochastic theory of diffusion-controlled reaction. *Journal of Physics A: Mathematical and General*, 9(9):1479, 1976.

[13] Aleksandar Donev, Eric Vanden-Eijnden, Alejandro Garcia, John Bell, et al. On the accuracy of finite-volume schemes for fluctuating hydrodynamics. *Communications in Applied Mathematics and Computational Science*, 5(2):149–197, 2010.

[14] Aleksandar Donev, Chiao-Yu Yang, and Changho Kim. Efficient reactive brownian dynamics. *The Journal of Chemical Physics*, 148(3):034103, 2018.

[15] Radek Erban and S Jonathan Chapman. Stochastic modelling of reaction–diffusion processes: algorithms for bimolecular reactions. *Physical biology*, 6(4):046001, 2009.

[16] Benjamin Fehrman and Benjamin Gess. Well-posedness of stochastic porous media equations with nonlinear, conservative noise. *arXiv preprint arXiv:1712.05775*, 2017.

[17] Peter G Fennell, Sergey Melnik, and James P Gleeson. Limitations of discrete-time approaches to continuous-time contagion dynamics. *Physical Review E*, 94(5):052125, 2016.

[18] Ronald Aylmer Fisher. The wave of advance of advantageous genes. *Annals of eugenics*, 7(4):355–369, 1937.

[19] Daniel T Gillespie. The chemical langevin equation. *The Journal of Chemical Physics*, 113(1):297–306, 2000.

[20] Daniel T Gillespie, Andreas Hellander, and Linda R Petzold. Perspective: Stochastic algorithms for chemical kinetics. *The Journal of chemical physics*, 138(17):05B201_1, 2013.

[21] Ana Grabundžija and Emmanuele Russo. Tools tell tales-climate trends changing threads in the prehistoric Pannonian Plain. *Documenta Praehistorica*, 43:301, 2016.

[22] Volker Grimm, Uta Berger, Finn Bastiansen, Sigrunn Eliassen, Vincent Ginot, Jarl Giske, John Goss-Custard, Tamara Grand, Simone K Heinz, Geir Huse, et al. A standard protocol for describing individual-based and agent-based models. *Ecological modelling*, 198(1-2):115–126, 2006.

[23] Volker Grimm, Uta Berger, Donald L DeAngelis, J Gary Polhill, Jarl Giske, and Steven F Railsback. The odd protocol: a review and first update. *Ecological modelling*, 221(23):2760–2768, 2010.

[24] Martin Hairer. An introduction to stochastic pdes. *arXiv preprint arXiv:0907.4178*, 2009.

[25] Dirk Helbing. Agent-based modeling, 2012.

[26] Desmond J Higham. An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM review*, 43(3):525–546, 2001.

[27] Wolfgang Hörmann, Josef Leydold, and Gerhard Derflinger. General principles in random variate generation. In *Automatic Nonuniform Random Variate Generation*, pages 13–41. Springer, 2004.

[28] Samuel A Isaacson. A convergent reaction-diffusion master equation. *The Journal of chemical physics*, 139(5):054101, 2013.

[29] Volker John. Numerical methods for partial differential equations. *Lecture notes*, 2013.

[30] William O Kermack and Anderson G McKendrick. A contribution to the mathematical theory of epidemics. In *Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences*, volume 115, pages 700–721. The Royal Society, 1927.

[31] Changho Kim, Andy Nonaka, John B Bell, Alejandro L Garcia, and Aleksandar Donev. Stochastic simulation of reaction-diffusion systems: A fluctuating-hydrodynamics approach. *The Journal of chemical physics*, 146(12):124110, 2017.

[32] Peter E Kloeden and Eckhard Platen. Higher-order implicit strong numerical schemes for stochastic differential equations. *Journal of statistical physics*, 66(1):283–314, 1992.

[33] Peter E Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*. Springer, 1992.

[34] Gabriel J Lord, Catherine E Powell, and Tony Shardlow. *An introduction to computational stochastic PDEs*. Number 50. Cambridge University Press, 2014.

[35] Michael W Macy and Robert Willer. From factors to actors: computational sociology and agent-based modeling. *Annual review of sociology*, 28(1):143–166, 2002.

[36] David Meintrup and Stefan Schäffler. *Stochastik: Theorie und Anwendungen*. Springer-Verlag, 2006.

[37] Bernt Øksendal. Stochastic differential equations. In *Stochastic differential equations*, pages 65–84. Springer, 2003.

[38] Martin Park, Nataša Djurdjevac Conrad, Ana Grabundzija, Luzie Helfmann, Emmanuele Russo, Marcus Weber, Johannes Zonker, Wolfram Schier, Christof Schütte, and Brigitta Schütt. Modeling the spread of the wool-bearing sheep from south-west asia into europe – an agent-based approach. Submitted.

[39] Romualdo Pastor-Satorras, Claudio Castellano, Piet Van Mieghem, and Alessandro Vespignani. Epidemic processes in complex networks. *Reviews of modern physics*, 87(3):925, 2015.

[40] Grigorios A Pavliotis. *Stochastic processes and applications*. Springer, 2016.

[41] John E Pearson. Complex patterns in a simple system. *Science*, 261(5118):189–192, 1993.

[42] Tobias Reichenbach, Mauro Mobilia, and Erwin Frey. Mobility promotes and jeopardizes biodiversity in rock–paper–scissors games. *Nature*, 448(7157):1046, 2007.

[43] C Tineke Rooijakkers. Spinning animal fibres at Late Neolithic Tell Sabi Abyad, Syria? *Paléorient*, pages 93–109, 2012.

[44] Christof Schütte and Marco Sarich. *Metastability and Markov state models in molecular dynamics: modeling, analysis, algorithmic approaches*, volume 24. American Mathematical Soc., 2013.

[45] Frank Schweitzer. Modelling migration and economic agglomeration with active brownian particles, 2002.

[46] Marian von Smoluchowski. Versuch einer mathematischen theorie der koagulationskinetik kolloider lösungen. *Zeitschrift für physikalische Chemie*, 92(1):129–168, 1918.

[47] Christian L Vestergaard and Mathieu Génois. Temporal gillespie algorithm: Fast simulation of contagion processes on time-varying networks. *PLoS computational biology*, 11(10):e1004579, 2015.

[48] Stefanie Winkelmann and Christof Schütte. The spatiotemporal master equation: Approximation of reaction-diffusion dynamics via markov state modeling. *The Journal of Chemical Physics*, 145(21):214107, 2016.