May 2013

Markov Chain Monte Carlo Simulation of the Wright-Fisher Diffusion

Markus Joseph Wahl
University of Wisconsin-Milwaukee

Markov Chain Monte Carlo Simulation of the Wright-Fisher Diffusion

by

Markus J. Wahl

A Thesis Submitted in
Partial Fulfillment of the
Requirements for the Degree of

Master of Science
in
Mathematics

at

The University of Wisconsin-Milwaukee
May 2013

ABSTRACT

Markov Chain Monte Carlo Simulation of the Wright-Fisher Diffusion

by

Markus J. Wahl

The University of Wisconsin-Milwaukee, 2013
Under the Supervision of Advisor
Professor Richard H. Stockbridge

In population genetics, the proportions of alleles at any given time are of interest. From generation to generation, these proportions vary and over a long time horizon the likelihoods for the proportions are given by a stationary distribution corresponding to the dynamics of the population. We investigate a diffusion approximation for the Wright-Fisher model and develop a Markov chain Monte Carlo simulation to approximate the evolution of the proportions of alleles in the population. Our aim is to estimate the stationary distribution, especially for parameters of the model for which no analytical formulas are known. We discretize the space of the diffusion process and construct a continuous time Markov chain which converges weakly to the diffusion.

Table of Contents

List of Figures

ACKNOWLEDGMENTS

First, I express my deepest gratitude to my thesis advisor Professor Richard Stockbridge for his guidance throughout this thesis project and the whole year at the University of Wisconsin-Milwaukee. I especially thank him for the opportunity to work on such an interesting project.

Furthermore, I thank Professor Eric Key and Professor Chao Zhu for serving on my committee, for several discussions and for revising my thesis.

Finally, I am very grateful for the support of my family.

Chapter 1

Introduction

In a population, in which certain genes can be of different types, called alleles, questions arise how the composition of the population will evolve over time. How will the population be built-up in the future? Will a certain allele become extinct? What proportions of the population are made up of each allele?

One mathematical model is provided by the Wright-Fisher diffusion. This model takes not only the randomness for the reproduction of individuals into account, but also mutation (the possibility that one gene changes from one allele to another) and selection (better chances of survival for certain genes of a certain allele).

The questions we described above considering the long-term behavior of the population correspond to the stationary distribution in the diffusion model. Stationary distributions for the Wright-Fisher diffusion are known only for a few special combinations of parameters. However, the long-term behavior is unknown for more general combinations of parameters.

Since no analytical formulas exist for these cases, we develop a Markov chain Monte Carlo algorithm to simulate the evolution of the population in order to approximate the stationary distribution of the Wright-Fisher diffusion model.

The diffusion is approximated by a discretization. We do not discretize the time

here as in other schemes used for the simulation of diffusion processes such as the Euler-Maruyama method. Instead, we discretize the state space to get an approximating continuous time Markov Chain with finite state space, using a rectangular grid, in our case. Then, we simulate the stationary distribution of the approximating Markov chain using Markov chain Monte Carlo techniques. Since the Markov chain is constructed as a locally consistent approximation, convergence in distribution of the simulated stationary distributions to the stationary distribution of the diffusion process is guaranteed as the grid becomes finer.

First of all, we describe the Markov chain Monte Carlo method for continuous time Markov chains in this chapter. To construct this Markov Chain, we use a combination of two methods. These methods are described in Chapter 2 according to [KD2001]. In Chapter 3, we describe the application of the techniques from Chapter 2 to the Wright-Fisher diffusion and their combination for the construction of the Markov chain Monte Carlo algorithm. Chapter 3 provides the theoretical basis for the algorithm. We discuss the numerical results of an implementation of this algorithm in Chapter 4.

1.1 Markov Chain Monte Carlo Simulation

Since we approximate the diffusion process by a continuous time Markov chain, the approximation of the stationary distribution of the diffusion process is the stationary distribution of the continuous time Markov chain. Thus, we simulate the unknown stationary distribution of the Markov chain. In this section, we describe the Markov chain Monte Carlo method used for the simulation of this distribution.

Let $(\psi(t))_{t\geq 0}$ be a continuous time Markov chain with finite state space S, transition matrices $(P(t))_{t\geq 0}$, with $P(t) = (p_{ij}(t))_{i,j\in S}$. We denote the embedded chain by ξ_n for $n \in \mathbb{N}$.

If ξ_n is irreducible and $(\psi(t))_{t\geq 0}$ is positive recurrent, then $(\psi(t))_{t\geq 0}$ has a stationary distribution $\pi = (\pi_i)_{i\in S}$ and

$$p_{ij}(t) \to \pi_j \ \ (t \to \infty) \tag{1.1}$$

independent of i [Br1999].

For an arbitrary initial distribution $\phi = (\phi_i)_{i \in S}$, we have

$$\mathbb{P}(\psi(t) = j) = \sum_{i \in S} p_{ij}(t)\, \phi_i \to \sum_{i \in S} \pi_j \phi_i = \pi_j, \ \ (t \to \infty). \tag{1.2}$$

Thus, we can approximate π by running the Markov chain for a certain amount of time T. Since π does not depend on the initial distribution ϕ we can choose an arbitrary initial distribution. T has to be chosen large enough to allow a sufficient number of transitions.
This result allows us to simulate π in the following way.

1. Generate a realization $\hat{\phi}$ of ϕ.
2. Generate one realization $\hat{\psi}$ of a path of the Markov chain ψ with $\hat{\psi}(0) = \hat{\phi}$.
3. Repeat steps 1 and 2.

Denote the realized paths by $\left(\hat{\psi_m}(t)\right)_{0 \le t \le T}$, $m = 1, \ldots, M$. By (1.2) and the strong law of large numbers,

$$\pi(i) \approx \mathbb{P}(\psi(T) = i) = \mathbb{E}\left[\mathbb{1}_{\{\psi(T)=i\}}\right] = \lim_{M \to \infty} \frac{1}{M} \sum_{m=1}^{M} \hat{\psi_m}(T)\, \mathbb{1}_{\left\{\hat{\psi_m}(T)=i\right\}} \tag{1.3}$$

for T sufficiently large, so that enough transitions are allowed.

To simulate an approximation of π using this method, we need to construct a continuous time Markov chain with an irreducible embedded chain. The irreducibility can be verified easily in our application. Furthermore, we need positive recurrence for the continuous time Markov chain itself. This process is positive recurrent since the state space is finite and so the irreducibility of the embedded chain implies recurrence (and positive recurrence) of the embedded chain and the continuous time Markov chain. Positive recurrence implies then the existence of a

stationary distribution and (1.1). By (1.2), this stationary distribution is unique. Consequently, it suffices to check that the embedded chain is irreducible and that the approximation converges weakly as the grid becomes finer. This will be discussed in the next chapter.

Chapter 2

Locally Consistent Markov Chain Approximations

In this chapter we want to approximate the diffusion process X by a sequence of continuous time Markov chains ψ_h in a way such that $\psi_h \longrightarrow X$ weakly for $h \longrightarrow 0$ as described in [KD2001]. Two issues arise here. The first issue is, which conditions does the approximation have to satisfy in order to get the desired behavior. This behavior is guaranteed for locally consistent approximations. Therefore, we will begin Chapter 2 by discussing local consistency.

The second issue is how to actually construct such a locally consistent Markov chain approximation for a diffusion process. We use a combination of two methods, which we describe in Chapter 2 as well.

2.1 Local Consistency

Let

$$Af(x) = \frac{1}{2}\sum_{i,j=1}^{d} a_{ij}(x)\frac{\partial^2}{\partial x_i \partial x_j} f(x) + \sum_{i=1}^{d} b_i(x)\frac{\partial}{\partial x_i} f(x) \tag{2.1}$$

be the generator of a d-dimensional diffusion process $X = (X(t))_{t\geq 0}$ which is to be approximated.

Definition 2.1.1 (Local Consistency)**.** A family of continuous time Markov chains ψ_h with finite state spaces $S_h \subset \mathbb{R}^d$, expected holding times $\Delta t^h(x) > 0,\ x \in S_h$

and embedded chain ξ_n^h is called a locally consistent approximation of the diffusion process X if

$$\sup_{x,\omega} \left\| \xi_{n+1}^h - \xi_n^h \right\| \to 0, \ \sup_x \Delta t^h(x) \to 0 \text{ as } h \to 0, \ \inf_x \Delta t^h(x) > 0 \text{ for each } h \quad (2.2)$$

and further

$$\mathbb{E}_x \left[\xi_{n+1}^h - \xi_n^h \right] = \Delta t^h(x)\, b(x) + o\left(\Delta t^h(x)\right) \quad (2.3)$$

and

$$\mathbb{E}_x \left[\left(\Delta\xi_n^h - \mathbb{E}\Delta\xi_n^h \right) \left(\Delta\xi_n^h - \mathbb{E}\Delta\xi_n^h \right)^T \right] = \Delta t^h(x)\, a(x) + o\left(\Delta t^h(x)\right), \quad (2.4)$$

where $\Delta\xi_n^h = \xi_{n+1}^h - \xi_n^h$.

These conditions ensure the desired weak convergence of the approximating process to the diffusion [KD2001]. Thus, we have to construct Markov chains which have the properties (2.2), (2.3) and (2.4). In our case, (2.2) will be obvious and holds always, whereas (2.3) and (2.4) turn out to be challenging. We therefore focus on (2.3) and (2.4) in our descriptions. We always use a uniform rectangular grid for S_h.

2.2 Construction of the Locally Consistent Approximating Markov Chain

The following two approaches from [KD2001] for the construction of the locally consistent approximating Markov chain are used for our algorithm. Whereas the first approach (in 2.2.1) uses finite difference techniques and is therefore a numerical approach, the secend approach (described in 2.2.2) is a probabilistic, direct approach to get the transition probabilities and transition rates.

2.2.1 Approximation using Finite Differences

The idea behind this approach is to approximate the partial derivatives from the generator of the diffusion process in (2.1) by finite differences in a specific way

such that the resulting operator can be interpreted as a generator of a continuous time Markov chain. Denote the points that are used for the difference quotient to approximate the partial derivatives at point x by $x+hv_i,\ i=1,...,K$. By choosing appropriate difference quotients and collecting terms, we want to write A as

$$Af(x) \approx c(x) \sum_{i=1}^{K} p_i(x) \left(f(x+hv_i) - f(x)\right) \tag{2.5}$$

in such a way that $p_i(x) \geq 0$ and $\sum_{i=1}^{K} p_i(x) = 1$. Then we can interpret (2.5) as an infinitesimal generator of a continuous time Markov chain, where $p^h(x, x+hv_i) := p_i(x),\ i=1,...,K$ are the transition probabilities of the embedded chain for the transition from x to $x+hv_i$ and $c(x)$ are the transition rates, given the chain is in state x.

The approximations of the partial derivatives are chosen as described below, where e_i are the canonical unit vectors:

$$f_{x_i}(x) \approx \begin{cases} \dfrac{f(x+he_i) - f(x)}{h}, & \text{if } b_i(x) \geq 0 \\ \dfrac{f(x) - f(x-he_i)}{h}, & \text{if } b_i(x) < 0 \end{cases}. \tag{2.6}$$

These different choices for the difference quotient ensure that $p^h(x \pm he_i)$ are positive. The geometric interpretation helps to understand this choice: In case $b_i(x) > 0$, this component of the drift points from x to $x+he_i$ (for $b_i(x) < 0$ from x to $x-he_i$). Thus, $f(x)$ and $f(x+he_i)$ are used in the first case and $f(x)$ and $f(x-he_i)$ in the second case for the difference quotient.

The reasons for the choices for the approximation of the second partial derivatives are the same, but the concern is not about the drift here but about the diffusion instead. A different difference quotient for $f_{x_ix_i}$ compared to what is usually used to compute the mixed partial derivatives is chosen for the same reason.

These choices are

$$f_{x_ix_i}(x) \approx \frac{f(x+he_i) + f(x-he_i) - 2f(x)}{h^2}, \tag{2.7}$$

for $a_{ij}(x) \geq 0$, we use

$$f_{x_i x_j}(x) \approx \frac{2f(x) + f(x + he_i + he_j) - f(x - he_i - he_j)}{2h^2} - \frac{f(x + he_i) + f(x - he_i) - f(x + he_j) + f(x - he_j)}{2h^2} \tag{2.8}$$

and for $a_{ij}(x) < 0$ we use

$$f_{x_i x_j}(x) \approx -\frac{2f(x) + f(x + he_i - he_j) + f(x - he_i + he_j)}{2h^2} + \frac{f(x + he_i) + f(x - he_i) - f(x + he_j) + f(x - he_j)}{2h^2}. \tag{2.9}$$

By approximating the derivatives in (2.1) by the difference quotients (2.6), (2.7), (2.8), (2.9) and rearranging the terms in the form (2.5), one obtains the transition probabilities for the embedded chain

$$\begin{aligned} p(x, x \pm he_i) &= \frac{\frac{a_{ii}(x)}{2} - \sum_{j:j\neq i} \frac{|a_{ij}(x)|}{2} + hb_i^{\pm}(x)}{Q^h(x)} \\ p(x, x + he_i + he_j) &= p(x, x - he_i - he_j) = \frac{a_{ij}^{+}(x)}{2Q^h(x)} \\ p(x, x - he_i + he_j) &= p(x, x + he_i - he_j) = \frac{a_{ij}^{-}(x)}{2Q^h(x)} \end{aligned} \tag{2.10}$$

where

$$Q^h(x) = \sum_i a_{ii}(x) - \sum_{i,j:i\neq j} \frac{|a_{ij}(x)|}{2} + h\sum_i |b_i(x)| \tag{2.11}$$

is a normalizing constant. This gives us the expected holding times

$$\Delta t^h(x) = \frac{h^2}{Q^h(x)}. \tag{2.12}$$

The quantities in (2.10) are nonnegative and sum up to one and can therefore be interpreted as probabilities as long as

$$a_{ii}(x) - \sum_{j:j\neq i} |a_{ij}(x)| \geq 0. \tag{2.13}$$

It is not obvious from the construction that these transition probabilities together with $\Delta t^h(x)$ give us a locally consistent Markov chain approximation. However, (2.3) and (2.4) can be verified [KD2001], so we have a locally consistent approximation with possible transition directions to the nearest neighbors parallel to the axes and the diagonals as illustrated in Figure 2.1. Both the uniform grid for S_h

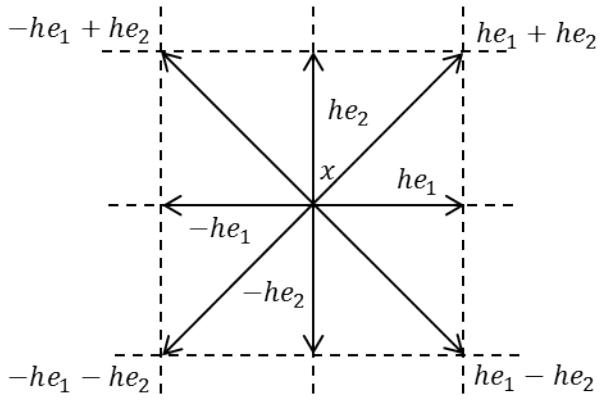

Figure 2.1: Transition directions

and the possible transition directions result from the choices for the difference quotients. That means the structure of the approximating chain is basically influenced by the points used for the difference quotients. Since these points can not be chosen arbitrarily, but instead they have to be chosen in a way such that the result can be interpreted as the generator of the Markov chain, it is hard to change these directions. We will see in Section 3.2.3 that it will be necessary in our application to modify these directions. To be able to adjust the algorithm in such a way, we describe the following approach, in which the possible transition directions are not the result of the finite difference approximation, but, instead, are chosen directly.

2.2.2 Direct Construction of the Approximating Markov Chain

This approach is a direct and therefore more probabilistic approach. Unlike the one just described, there is no discretization of the generator here. Instead, the

approach aims at directly finding transition probabilities for an embedded chain and transition rates such that the local consistency conditions (2.3) and (2.4) hold. This is not trivial since both the expectation (drift) and the covariance structure must match. To simplify this task, in this approach, we decompose the transition into a drift and a diffusion component and then combine both components.

We choose for each x a set $M(x) = \{v_i(x),\ i \leq m(x)\},\ m(x) \leq K$ of vectors $v_i(x)$ as candidates for the next states of the chain. For the local consistency, the transition probabilities must be chosen such that they satisfy (2.3) and (2.4), that is

$$\sum_{i \in M(x)} p(x, x + hv_i(x))\, hv_i(x) = \Delta t^h(x)\, b(x) + o\left(\Delta t^h(x)\right) \tag{2.14}$$

for (2.3) and

$$\sum_{i \in M(x)} p(x, x + hv_i(x))\, hv_i(x)\, hv_i(x)^T = a(x)\, \Delta t^h(x) + o\left(\Delta t^h(x)\right). \tag{2.15}$$

for (2.4).

The idea is initially to deal with these two equations separately. We want to find $q_i^0(x) \geq 0$ and $q_i^1(x) \geq 0$ for all $i \in \{1, \ldots, m(x)\}$ and for all x such that

$$b(x) = \sum_{i \in M(x)} q_i^0(x)\, v_i(x) \tag{2.16}$$

and

$$a(x) = \sum_{i \in M(x)} q_i^1(x)\, v_i(x)\, v_i(x)^T. \tag{2.17}$$

Thus, $q_i^0(x)$ approximates the drift consistently and $q_i^1(x)$ approximates the diffusion. Since we handle both parts separately, we do not want the diffusion to contribute to the drift so we require further

$$\sum_{i \in M(x)} q_i^1 v_i(x) = 0. \tag{2.18}$$

By combining $q_i^0(x)$ and $q_i^1(x)$ we define the transition probabilities for the embedded chain by

$$p(x, x + hv_i(x)) = \frac{hq_i^0(x) + q_i^1(x)}{Q^h(x)} \tag{2.19}$$

with normalizing constant

$$Q^h(x) = \sum_{i \in M(x)} \left[hq_i^0(x) + q_i^1(x) \right]. \tag{2.20}$$

The corresponding holding times for these transition probabilities are then given by

$$\Delta t^h(x) = \frac{h^2}{Q^h(x)}. \tag{2.21}$$

Chapter 3

Markov Chain Approximation for the Wright-Fisher Diffusion

3.1 The Wright-Fisher Model

The Wright-Fisher model describes the evolution of the proportions of genes in a certain population. These genes can be of r different types (alleles). For the construction of the approximating Markov chain in Section 3.2, we consider the case $r = 3$. From each generation to the next, the model describes the change in the proportions of the alleles of the whole population by taking mutation, selection and randomness for the type of offsprings into account. The resulting process is a Markov chain. It is in many cases more convenient to work with a diffusion approximation, the so called Wright-Fisher diffusion, which we describe as in [EK1986].

3.1.1 The Wright-Fisher Diffusion

Let r be the number of possible alleles and $(X_1(t), \ldots, X_r(t))$ the process with $X_i(t)$ describing the proportion of allele i at time t, $t \geq 0$. Therefore, $\sum_{i=1}^{r} X_i(t) = 1$. The Wright-Fisher diffusion captures this condition by modeling $X = (X(t))_{t\geq 0}$, $X(t) = (X_1(t), \ldots, X_{r-1}(t))$ by a $r-1$ dimensional diffusion process on

$$K = \left\{ x = (x_1, \ldots, x_{r-1}) \in [0,1]^{r-1} : \sum_{i=1}^{r-1} x_i \leq 1 \right\}. \tag{3.1}$$

Then, $X_r(t) = 1 - \sum_{i=1}^{r-1} X_i(t)$.

We consider a version with mutation, but without selection. The diffusion operator of X for $f \in C^2(K)$ is given by

$$Af(x) = \frac{1}{2} \sum_{i,j=1}^{r-1} a_{ij}(x) \frac{\partial^2}{\partial x_i \partial x_j} f(x) + \sum_{i=1}^{r-1} b_i(x) \frac{\partial}{\partial x_i} f(x), \tag{3.2}$$

where

$$a_{ij}(x) = x_i(\delta_{ij} - x_j) \tag{3.3}$$

and

$$b_i(x) = -\sum_{j=1}^{r} \mu_{ij} x_i + \sum_{j=1}^{r} \mu_{ji} x_j. \tag{3.4}$$

3.2 Construction of the Approximating Markov Chain

3.2.1 Finite Difference Approximation Scheme for the Interior

We use the scheme obtained from the finite difference discretization described in (2.2.1) wherever possible. To verify (2.13) we observe that for $k \neq i$

$$a_{ii}(x) - \sum_{j:j\neq i} |a_{ij}(x)| = x_i(1 - x_i) - x_i x_k = x_i(1 - x_i - x_k) \geq 0 \tag{3.5}$$

since $x_i,\ x_k \in K$.

Consequently, we can use this algorithm in the interior. This gives us an algorithm with possible transitions to the nearest neighbors parallel to the axis and to the nearest diagonal neighbors. Note that $p(x, x + he_i + he_j) = p(x, x - he_i - he_j) = \frac{a_{ij}^+(x)}{2Q^h(x)} = 0$ since we have $a_{ij}(x) = -x_i x_j \leq 0$ for $i \neq j$ for all x. That means we only move along one diagonal and from the grid point x never directly to the neighbors $x + he_i + he_j$ and $x - he_i - he_j$.

We can apply this algorithm everywhere where there is no danger of leaving K. Since the transition directions are limited to the ones shown in Figure 3.1 and there

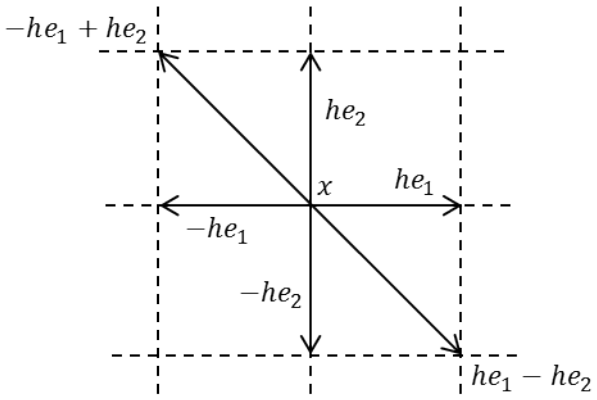

Figure 3.1: Possible transition directions

are no transitions along the other diagonal, we can use this algorithm on the grid points below the diagonal, i.e. for all $x = (x_1, x_2)$ with $x_1 + x_2 = 1 - h$ as well. This fact enables us to use the algorithm on the whole interior.

3.2.2 Approximation on the Horizontal and Vertical Boundary

Next, we want to show that we can use this scheme on the horizontal and vertical boundaries as well, i.e. where $x_1 = 0$ or $x_2 = 0$. That means we need to show that the probability of leaving the triangle, which is $p(x, x - he_1)$ is 0. Let us consider the case $x_1 = 0$ and $x_2 \neq 0, 1$, the argument for $x_2 = 0$ is essentially the same. For $x_1 = 0$ we have

$$\begin{aligned} p(x, x - he_1) &= \frac{\frac{a_{11}(x)}{2} - \frac{|a_{12}(x)|}{2} + hb_i^-(x)}{Q^h(x)} \\ &= \frac{\frac{x_1(1-x_1)}{2} - \frac{x_1 x_2}{2} + h\left(-\sum_{j=1}^r \mu_{ij} x_1 + \sum_{j=1}^r \mu_{ji} x_j\right)^-}{Q^h(x)} = 0 \end{aligned} \tag{3.6}$$

since $\mu_{j,i} \geq 0$ for all $i, j = 1, \ldots, r$. Thus, we can use the the algorithm on these boundaries (except for the vertices) as well.

3.2.3 Approximation Scheme on the Diagonal

Now, we want to consider the diagonal, i.e. all $x = (x_1, x_2)$ for which $x_1 + x_2 = 1$. That means in particular, we include the vertices as well. However, the behavior at the vertices is slightly different since $a_{11} = a_{12} = 0$ at the vertices. As we will see, we can interprete the vertices both as special cases of points on the diagonal and as different cases with no diffusion. First, we include the vertices in the diagonal. Although the Wright-Fisher diffusion never leaves K, this may happen to the discretization with positive probability as the following example describes.

Example 3.2.1. Let $\mu_{ij} = \mu_j = 0.5,\ i, j = 1, 2, 3$. Then we have on the diagonal, i.e. for $x_1 = 1 - h$ and $x_2 = h$

$$\begin{aligned} p(x, x + he_2) &= \frac{\frac{a_{22}(x)}{2} - \frac{|a_{21}(x)|}{2} + hb_2^+(x)}{Q^h(x)} \\ &= \frac{\frac{x_2(1-x_2)}{2} - \frac{x_2 x_1}{2} + h\left(-\sum_{j=1}^r \mu_j x_2 + \mu_2\right)^+}{Q^h(x)} \\ &= \frac{\frac{x_2(1-x_2-x_1)}{2} + h(-1.5h + 0.5)^+}{Q^h(x)} > 0 \end{aligned} \tag{3.7}$$

for $h < 1/3$, since $x_1 + x_2 = 1$ for all points on the diagonal.

As the Wright-Fisher diffusion never leaves the triangle, we have to modify the approximation on the diagonal. As a result we want to get transition probabilities and the expected holding times $\Delta t^h(x)$ such that the modified process is still locally consistent, but never leaves the triangle. The challenge is to maintain the local consistency without using the transition directions that point out of the triangle.

To solve this problem, let us first consider the reason for the problem. As (3.7) shows, the positive probability of leaving the triangle K results from the drift component, not from the diffusion component since $1 - x_1 - x_2 = 0$. The drift direction $b(x)$ is separated into its components $b_1(x)$ and $b_2(x)$. Although the total drift direction $b(x)$ points inside the triangle and the Wright-Fisher diffusion is instantly pulled back into the triangle once it approaches the diagonal, this approximating

process may leave the triangle, since the diffusion is approximated by the transitions in the horizontal and vertical direction and not by the diagonal. Figure 3.2 illustrates this. Here, $b_2(x) > 0$ and hence, $p(x, x + he_2) > 0$, but $x + he_2 \notin K$.

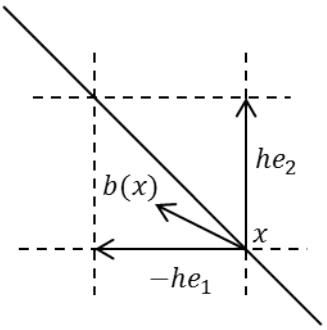

Figure 3.2: Leaving the triangle by a transition along he_2

Due to the nature of the Wright-Fisher diffusion, the drift at point x, where x is supposed to be on the diagonal, can never point out of the triangle, so we can assume

$$b_1(x) + b_2(x) \leq 0 \tag{3.8}$$

for the following solution of the problem described above. Furthermore, if both $b_1(x) \leq 0$ and $b_2(x) \leq 0$, the drift is decomposed to transition probabilities in the directions $-he_1$ and $-he_2$, so the process will never leave the triangle from such a point, as illustrated in Figure 3.3.

This alows us to use the original algorithm for the points with $b_1(x) \leq 0$ and $b_2(x) \leq 0$ so we can assume for the modification for the case pointed out in Figure 3.2 that either

$$b_1(x) < 0,\ b_2(x) > 0 \tag{3.9}$$

or

$$b_1(x) > 0,\ b_2(x) < 0 \tag{3.10}$$

holds.

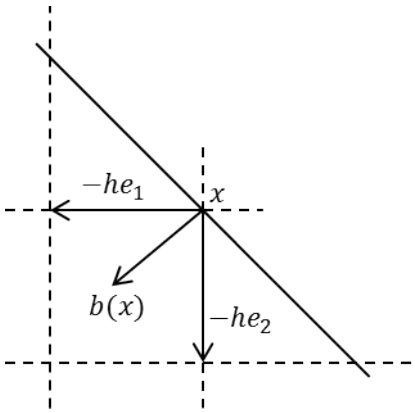

Figure 3.3: The algorithm can be used if $b_1(x) \leq 0$ and $b_2(x) \leq 0$.

Due to the symmetry, both cases are similar, so we assume in the following that (3.8) and (3.9) hold. In case that (3.8) and (3.10) hold, the derivation is then analogous. All other cases are not relevant as shown since they can not occur or they are covered by the algorithm described above.

3.2.4 Approximation on the Diagonal by Direct Construction

To find a locally consistent approximation on the diagonal, we use the method described in Section 2.2.2. First of all, we choose vectors $v_i,\ i = 1, 2, 3, 4$ as possible transition directions with

$$v_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix},\ v_2 = \begin{pmatrix} -1 \\ 0 \end{pmatrix} \text{ and } v_3 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{3.11}$$

The reasons for these choices are that this helps us on the one hand to replicate the drift by using v_1 and v_2. On the other hand, v_1 and v_3 are used to replicate the diffusion, which makes the process move only along the diagonal once it is on the diagonal. Next, we must choose $q_1^0(x)$ and $q_2^0(x)$ according to (2.16) such that

$$b(x) = \begin{pmatrix} b_1(x) \\ b_2(x) \end{pmatrix} = q_1^0(x) \begin{pmatrix} -1 \\ 1 \end{pmatrix} + q_2^0(x) \begin{pmatrix} -1 \\ 0 \end{pmatrix}. \tag{3.12}$$

These conditions force $q_1^0(x) = b_2(x) \geq 0$ by (3.9) and
$q_2^0(x) = -b_1(x) - b_2(x) \geq 0$ by (3.8).

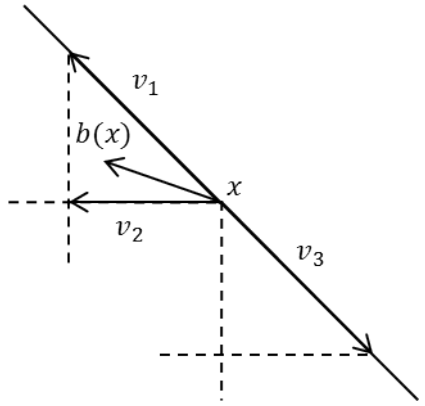

Figure 3.4: Transition directions $v_i,\ i = 1, 2, 3, 4$.

To find $q_1^1(x)$ and $q_3^1(x)$, we have to solve (2.17)

$$\begin{aligned}
&\begin{pmatrix} a_{11}(x) & a_{12}(x) \\ a_{21}(x) & a_{22}(x) \end{pmatrix} = \begin{pmatrix} x_1x_2 & -x_1x_2 \\ -x_1x_2 & x_1x_2 \end{pmatrix} \\
&= q_1^1(x) \begin{pmatrix} -1 \\ 1 \end{pmatrix} \begin{pmatrix} -1 & 1 \end{pmatrix} + q_3^1(x) \begin{pmatrix} 1 \\ -1 \end{pmatrix} \begin{pmatrix} 1 & -1 \end{pmatrix} \\
&= q_1^1(x) \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} + q_3^1(x) \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}.
\end{aligned} \tag{3.13}$$

Since the idea of this approach is the decomposition into drift and diffusion, the diffusion should not contribute to the drift, so we want $q_1^1(x)$ and $q_3^1(x)$ to further satisfy (2.18):

$$q_1^1(x) \begin{pmatrix} -1 \\ 1 \end{pmatrix} + q_3^1(x) \begin{pmatrix} 1 \\ -1 \end{pmatrix} = 0 \tag{3.14}$$

Using this result we obtain $q_1^1(x) = q_3^1(x)$ and so we can solve (3.13) which gives us

$$q_1^1(x) = q_3^1(x) = \frac{x_1x_2}{2}. \tag{3.15}$$

Then, we can set up the transition probabilities for the embedded chain and expected holding times and we have with (2.19)

$$\begin{aligned} p\left(x, x+hv_1\right) &= \frac{2hb_2\left(x\right)+x_1x_2}{2Q^h\left(x\right)}, \\ p\left(x, x+hv_2\right) &= -\frac{hb_1\left(x\right)+hb_2\left(x\right)}{Q^h\left(x\right)} \text{ and} \\ p\left(x, x+hv_3\right) &= \frac{x_1x_2}{2Q^h\left(x\right)} \end{aligned} \tag{3.16}$$

with the normalizing constant

$$Q^h\left(x\right) = -hb_1\left(x\right)+x_1x_2 \tag{3.17}$$

and expected holding times

$$\Delta t^h\left(x\right) = \frac{h^2}{Q^h\left(x\right)}. \tag{3.18}$$

due to (2.20) and (2.21).

3.2.5 Alternative Approximation Scheme for the Vertices

Consider the vertex $x_1 = 0,\ x_2 = 1$. The case $x_1 = 1,\ x_2 = 0$ is similar. For $x_1 = 0,\ x_2 = 1$, it follows for the diffusion that $a\left(x\right) = 0_{2\times 2}$. Thus, there is no diffusion at the vertex, the process is only pulled back by the drift. This behavior of the diffusion can be interpreted as deterministic. The idea behind this alternative approach is to obtain a locally consistent approximation by a deterministic transition. Local consistency means in this case that the drift has to be replicated by a transition to only one candidate, so the transition direction must fit the direction of the drift. In other words, that means that we proceed from x to a point on the grid which can be written as $x + \Delta t^h\left(x\right) b\left(x\right)$ for a small $\Delta t^h\left(x\right)$ which can be chosen appropriately. Of course, it is not always possible to find such a point which has to be positioned at the intersection of the grid and the drift vector starting at the vertex.

Example 3.2.2. Consider again the previous example with $\mu_i = 0.5,\ i = 1, 2, 3$. For $x = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, we have

$$b(x) = \begin{pmatrix} b_1(x) \\ b_2(x) \end{pmatrix} = \begin{pmatrix} -1 \\ 0.5 \end{pmatrix} = \Delta t^h(x) \begin{pmatrix} -2h \\ h \end{pmatrix} \tag{3.19}$$

with $\Delta t^h(x) = h/2$. Thus, we can use a deterministic transition from $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ to $\begin{pmatrix} 1-2h \\ h \end{pmatrix}$ with $\Delta t^h(x) = h/2$ to get a locally consistent approximation (see also figure 3.5).

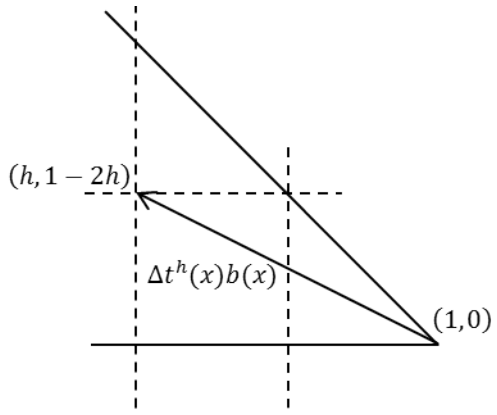

Figure 3.5: Deterministic transition at the vertex.

By construction, (2.3) holds and (2.4) holds as well, since $a_{i,j} = 0,\ i, j = 1, 2$ and the transition is deterministic. Thus, this deterministic transition at the vertex is locally consistent.

Chapter 4

Simulation Results

We describe the results of simulations of different types of stationary distributions in this chapter using the algorithm developed in the previous chapter. At the beginning, we approximate stationary distributions of the Wright-Fisher model choosing parameters for $\mu = (\mu_{ij})_{i,j=1,2,3}$ for which analytical formulas are known such that we can compare these simulation results to the distribution that is approximated. Afterwards, we simulate a distribution for parameters μ for which no analytical formula is known. The implementation of this algorithm can be found in the appendix.

Analytical formulas for the density of the stationary distribution of the Wright-Fisher diffusion are known if $\mu_{ij} = \mu_j$ for all $i, j = 1, \ldots, r,\ i \neq j$ and $\mu_{ii} = 0$. The density function is then given by the Dirichlet distribution

$$f(x_1, \ldots, x_{r-1}) = C \left(\prod_{i=1}^{r-1} x_i^{2\mu_i - 1} \right) \left(1 - \sum_{i=1}^{r-1} x_i \right)^{2\mu_r - 1} \tag{4.1}$$

with normalizing constant C [EK1986]. As a special case we have a uniform distribution when $\mu_j = 0.5$ for all $j = 1, \ldots, r$.

We always choose a uniform distribution on all grid point as initial distribution ϕ and the number of sample paths is always 100,000 throughout the whole chapter.

We deal with two types of convergence here. For $t \to \infty$, the distribution of $\psi(t)$, which is the distribution of the continuous time Markov chain at time t, converges to its stationary distribution π almost surely and therefore in L^1 as well as discussed

in (1.2), since the state space is finite. In order to observe this convergence, we run simulations until different times $T = 0.5, 1, 5$. $T = 5$ turned out to be an appropriate value in order to allow enough transitions as Figures (4.1)-(4.3) indicate. These figures show the distribution of the number of transitions made until $T = 5$ for the simulation from Section 4.2. The numbers of transition for the simulations from the other sections are similar.

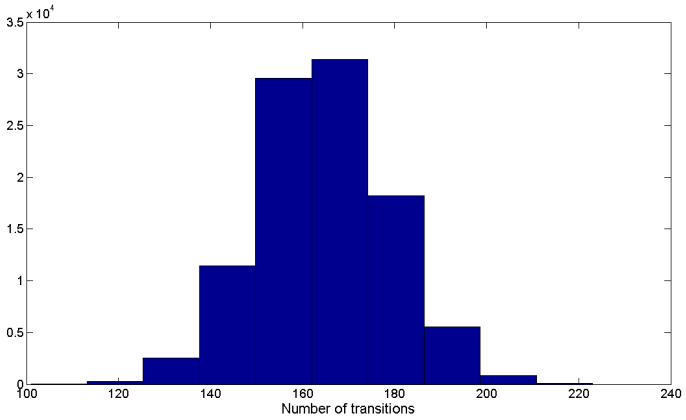

Figure 4.1: Distribution of the number of transitions for $h = 1/10$

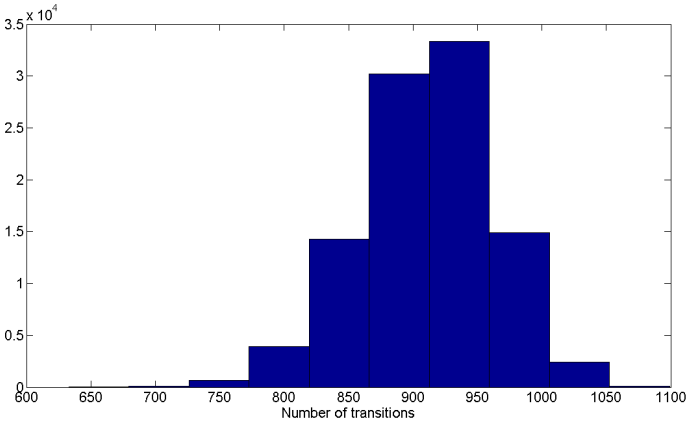

Figure 4.2: Distribution of the number of transitions for $h = 1/25$

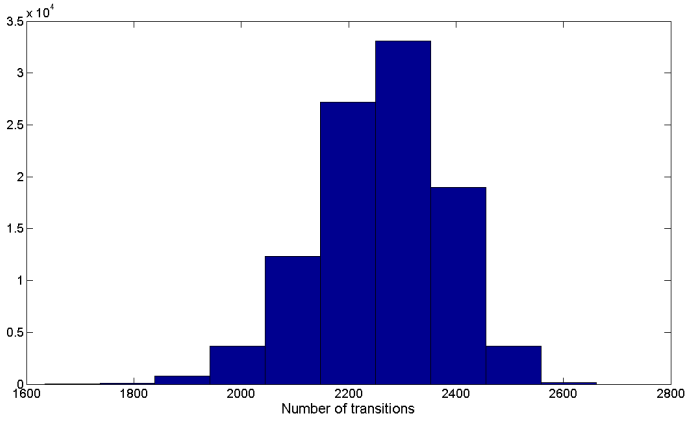

Figure 4.3: Distribution of the number of transitions for $h = 1/40$

The other type of convergence we deal with is the weak convergence of π to the stationary distribution of the diffusion as $h \to 0$, guaranteed by the local consistency [KD2001]. We investigate this convergence by running simulations for different grid sizes ($h = 1/10,\ 1/25$ and $1/40$).

4.1 Simulating Distributions with Known Analytical Formulas

The biological interpretation of μ helps us to understand the corresponding stationary distributions of the diffusion. Although mutation is not deterministic in nature, the parameters for the mutation, μ, influence the drift, but not the diffusion. The quantitiy μ_{ij} models the mutation from allele i to allele j. High values for μ_{ij} indicate a lot of mutation from allele i to allele j. Since an allele can not mutate to itself, $\mu_{ii} = 0$ is always required.

4.1.1 Uniform Distribution

First of all, we simulate the stationary distribution choosing

$$\mu = \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix}. \tag{4.2}$$

For this choice, the stationary distribution of the Wright-Fisher diffusion is the uniform distribution on K. Figures 4.4, 4.5 and 4.6 show that the simulated stationary distributions approximate the uniform distribution better for smaller h. Whereas the distribution in Figure 4.4 is shaped, the distribution in Figure 4.6 is almost flat. The convergence on the diagonal is slower due to the different discretization scheme which is not used in the middle of the diagonal, but near the vertices.

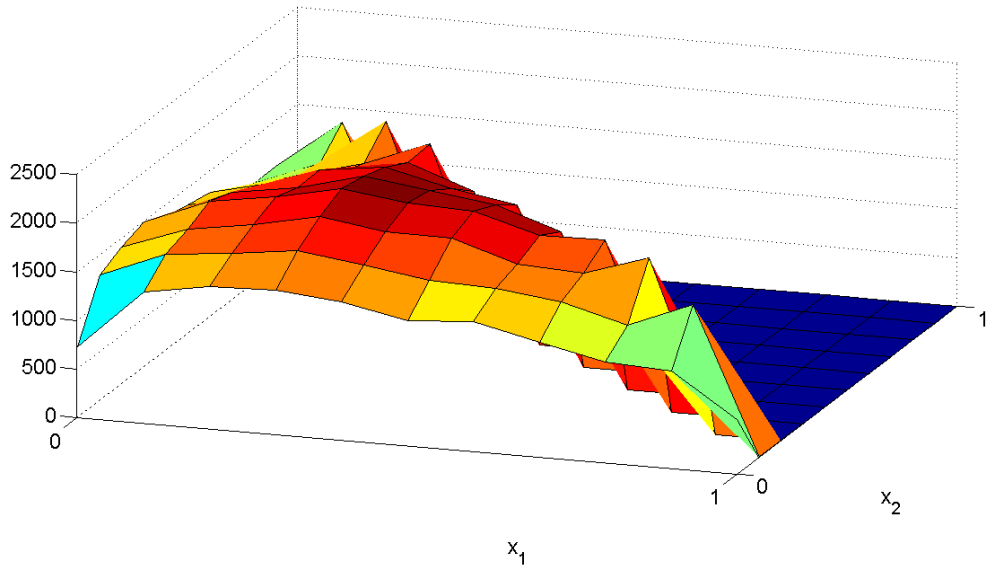

Figure 4.4: Simulation of the uniform distribution for $h = 1/10$, $T = 5$

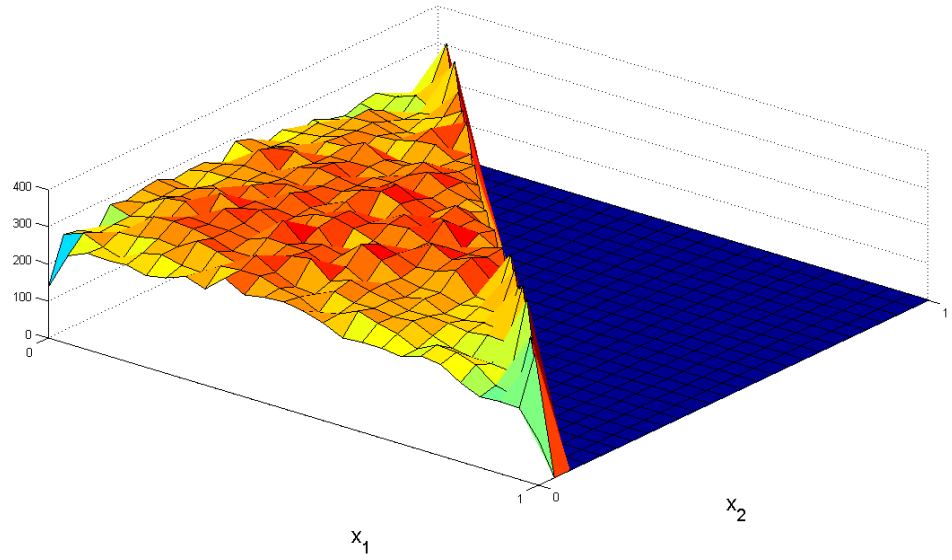

Figure 4.5: Simulation of the uniform distribution for $h = 1/25$, $T = 5$

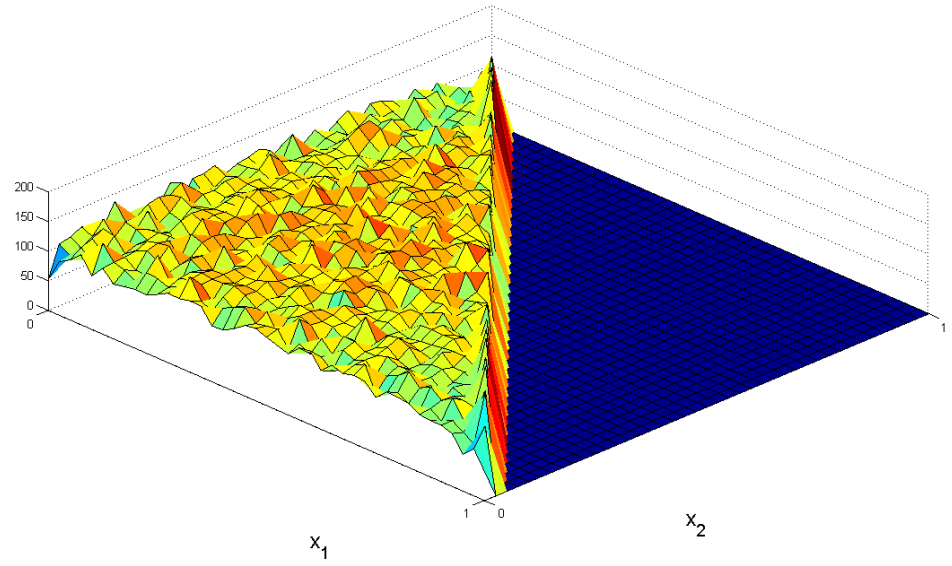

Figure 4.6: Simulation of the uniform distribution for $h = 1/40$, $T = 5$

4.1.2 Dirichlet Distribution

The next distribution we simulate is for

$$\mu = \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.3 & 0 & 0.5 \\ 0.3 & 0.5 & 0 \end{pmatrix}. \tag{4.3}$$

Compared to the uniform distribution, there is less mutation from alleles 2 and 3 to allele 1, since $\mu_{21} = \mu_{31} = 0.3$ compared to 0.5 in the previous case. Consequently, there is less probability mass concentrated on the larger proportions for allele 1. The stationary distribution of the diffusion is a Dirichlet distribution in this case. The density is given by

$$f(x_1, x_2) = 0.96 x_1^{-0.4} \tag{4.4}$$

for $x \in K$. Again, for $h = 1/10$, the curve is shaped parallel to the x_2 axis, even for $T = 5$ and the structure parallel to the x_1 axis is not similar to the Dirichlet distribution (Figures 4.7, 4.8 and 4.9). For $h = 1/25$ and $h = 1/40$, the simulations illustrate the convergence of the Markov chain approximation to the Dirichlet distribution. Furthermore, the transition from the initial, uniform distribution to the stationary distribution can be seen for these simulation runs in Figures 4.10-4.15. For $T = 0.5$ (Figures 4.10 and 4.13), much more probability mass is concentrated on the points with $x_1 >> 0$ and there are less realisations with small values for x_1 than for $T = 5$ (Figures 4.12 and 4.15).

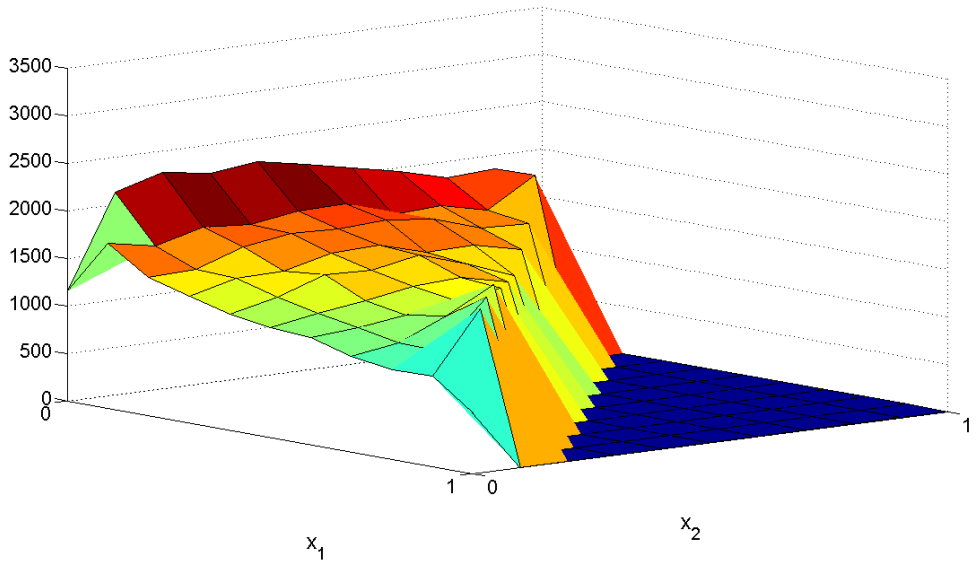

Figure 4.7: Simulation of the Dirichlet distribution with $h = 1/10$, $T = 0.5$

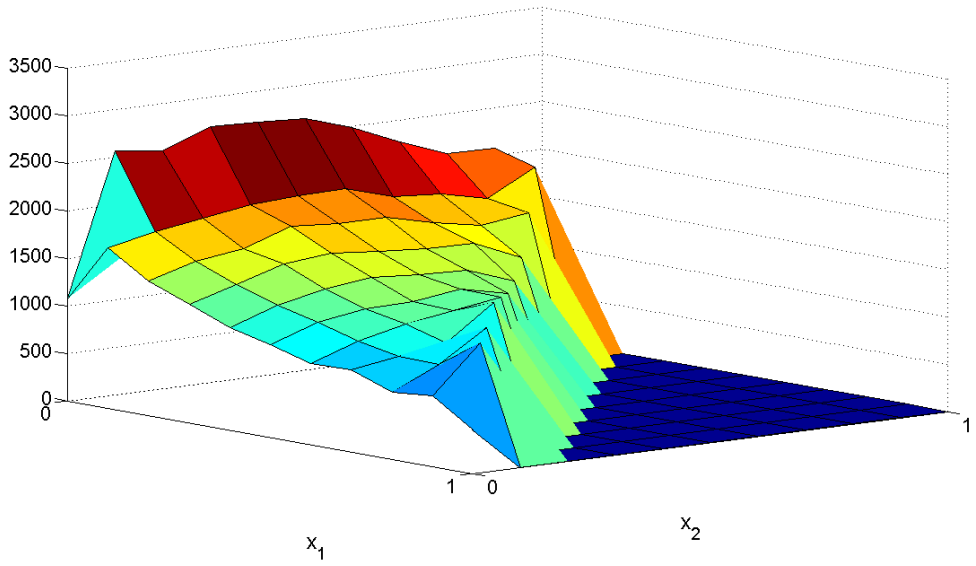

Figure 4.8: Simulation of the Dirichlet distribution with $h = 1/10$, $T = 1$

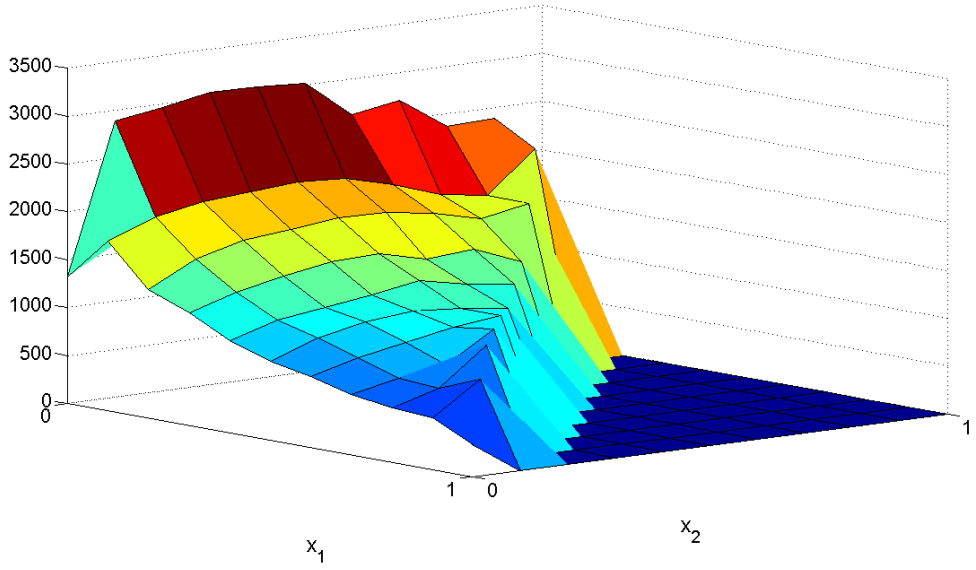

Figure 4.9: Simulation of the Dirichlet distribution with $h = 1/10$, $T = 5$

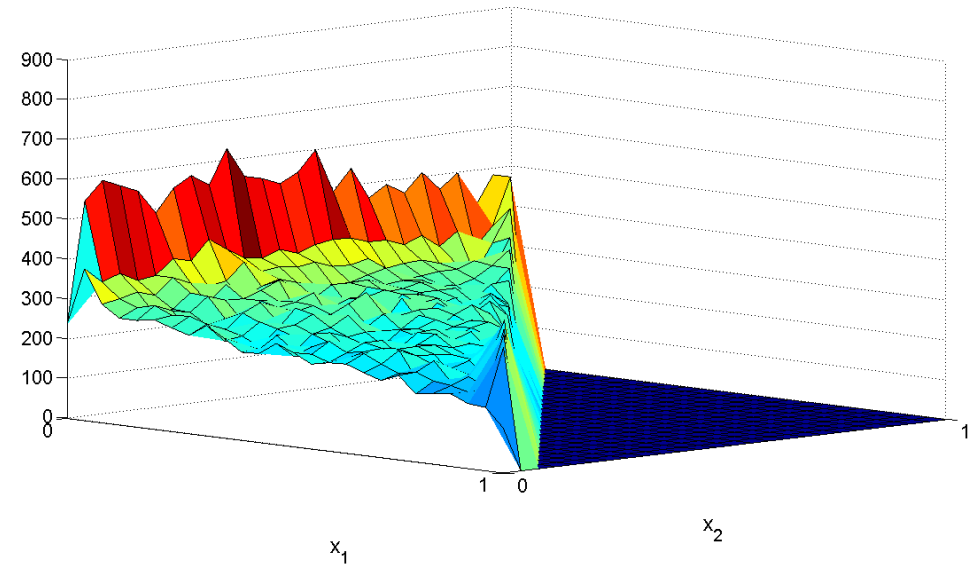

Figure 4.10: Simulation of the Dirichlet distribution with $h = 1/25$, $T = 0.5$

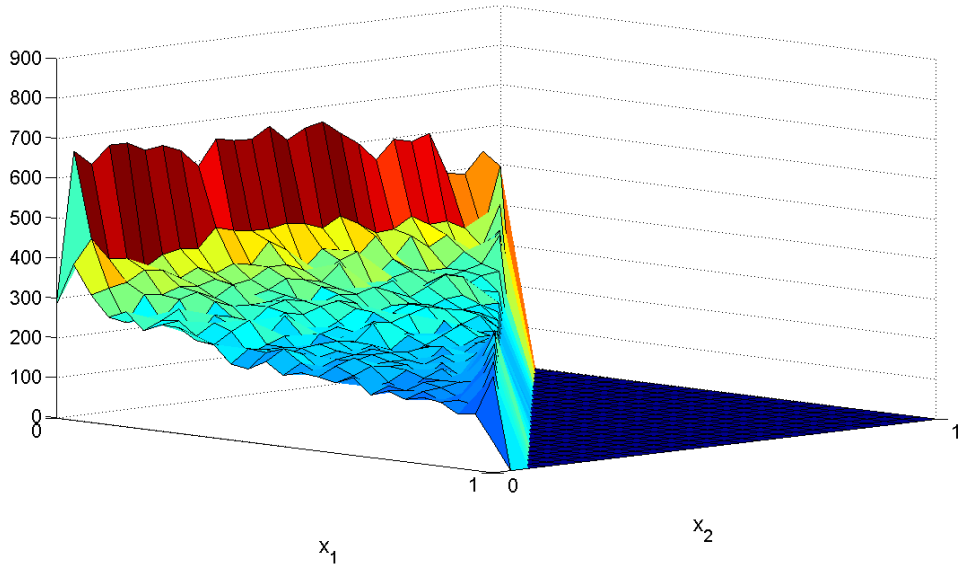

Figure 4.11: Simulation of the Dirichlet distribution with $h = 1/25$, $T = 1$

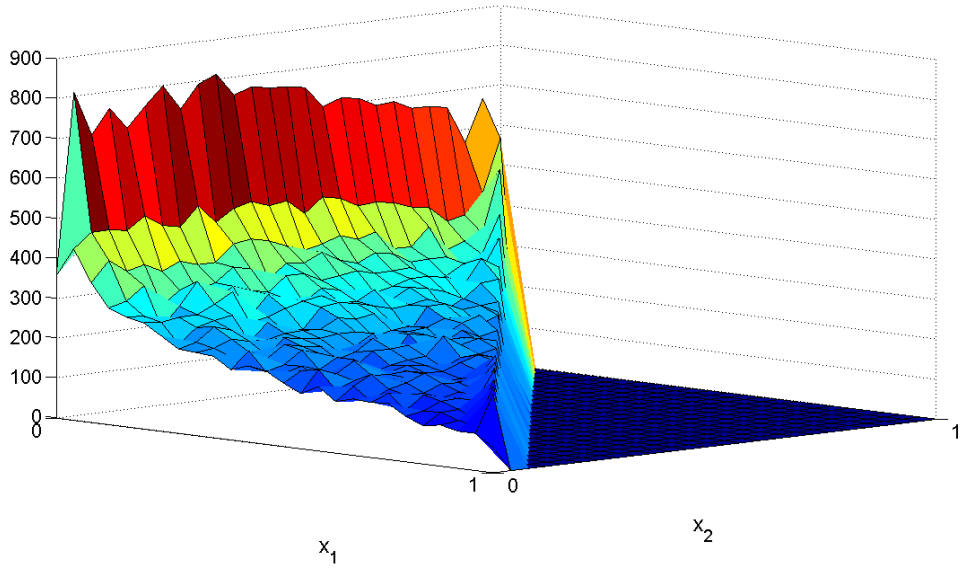

Figure 4.12: Simulation of the Dirichlet distribution with $h = 1/25$, $T = 5$

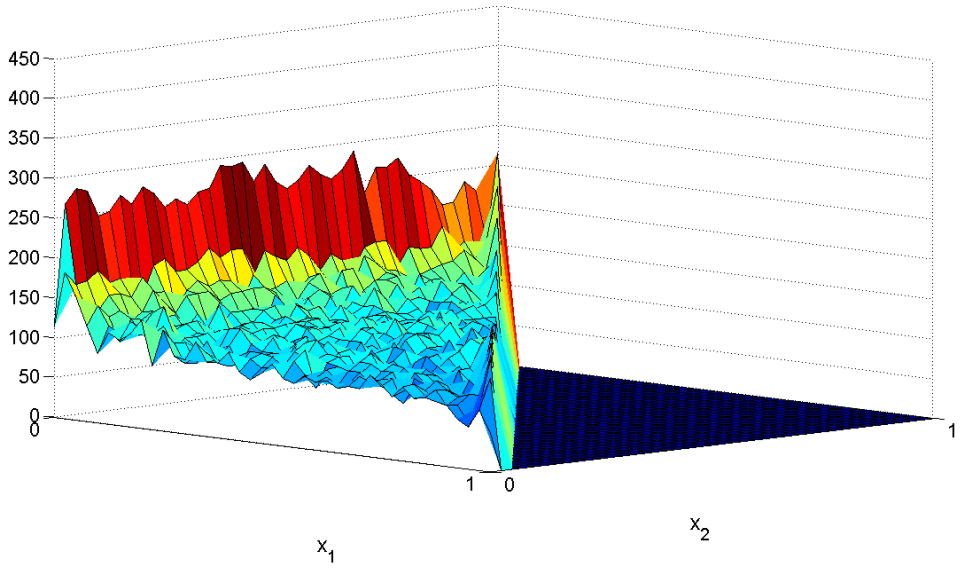

Figure 4.13: Simulation of the Dirichlet distribution with $h = 1/40$, $T = 0.5$

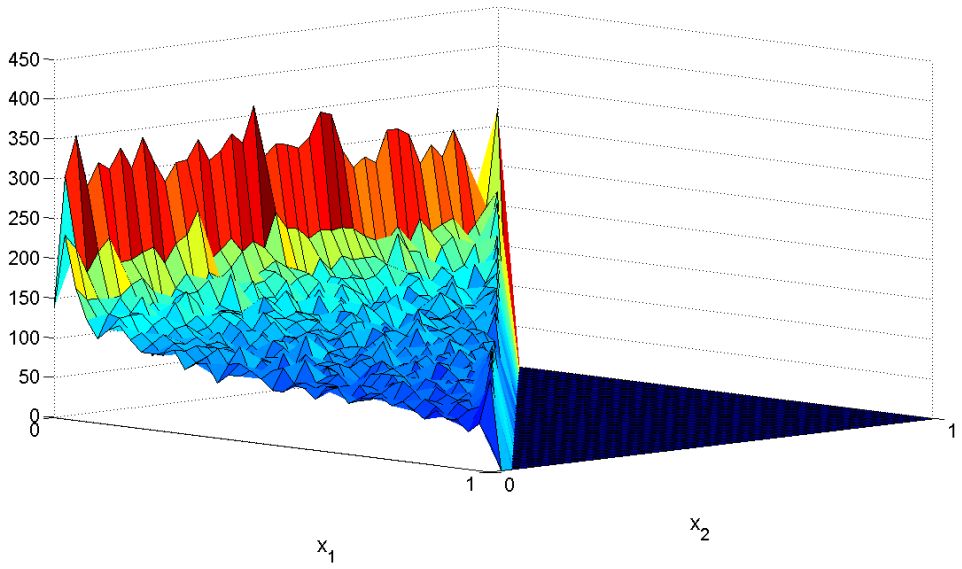

Figure 4.14: Simulation of the Dirichlet distribution with $h = 1/40$, $T = 1$

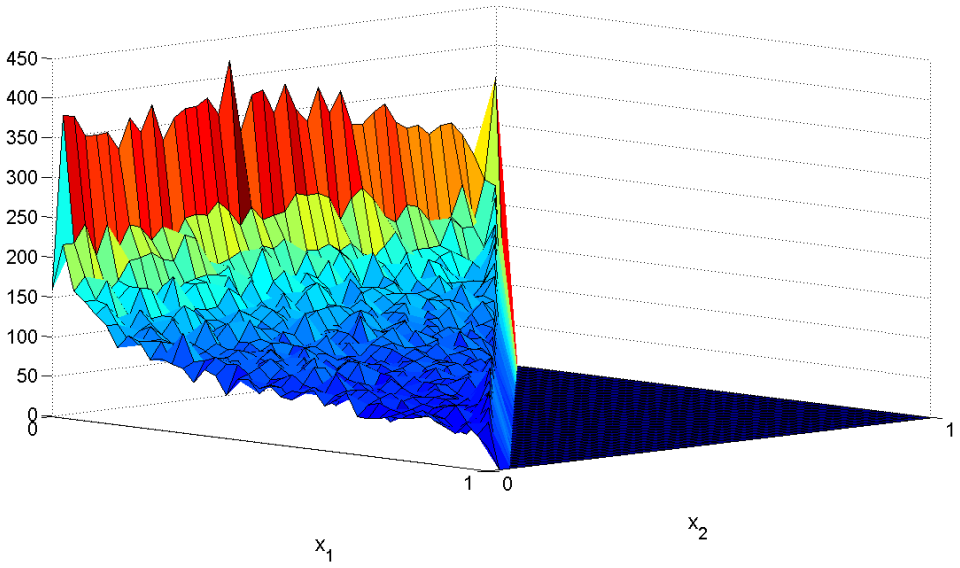

Figure 4.15: Simulation of the Dirichlet distribution with $h = 1/40$, $T = 5$

4.2 Simulations of a Distribution with Unknown Density

Finally, we want to simulate a stationary distribution of the Wright-Fisher diffusion with an unknown density. No analytical formula is known for the stationary distribution when

$$\mu = \begin{pmatrix} 0 & 0.5 & 1 \\ 0.5 & 0 & 1 \\ 0.5 & 1 & 0 \end{pmatrix}. \tag{4.5}$$

Compared to the uniform distribution, the higher value for μ_{13}, μ_{23} and μ_{32} corresponds to more mutation. Since mutation makes the process move away from the vertices, more probability mass is concentrated in the center of K. The likelihood to get combinations with small x_1, that is a small proportion of allele 1 is high, because the first column of μ (which is the mutation to allele 1) consists of lower values than the other columns.

Similar to the simulations of the Dirichlet distribution, the transition from the initial

distribution to the stationary distribution with more probability mass being concentrated in the center is clearly visible for all chosen values of h (Figures 4.16-4.24).

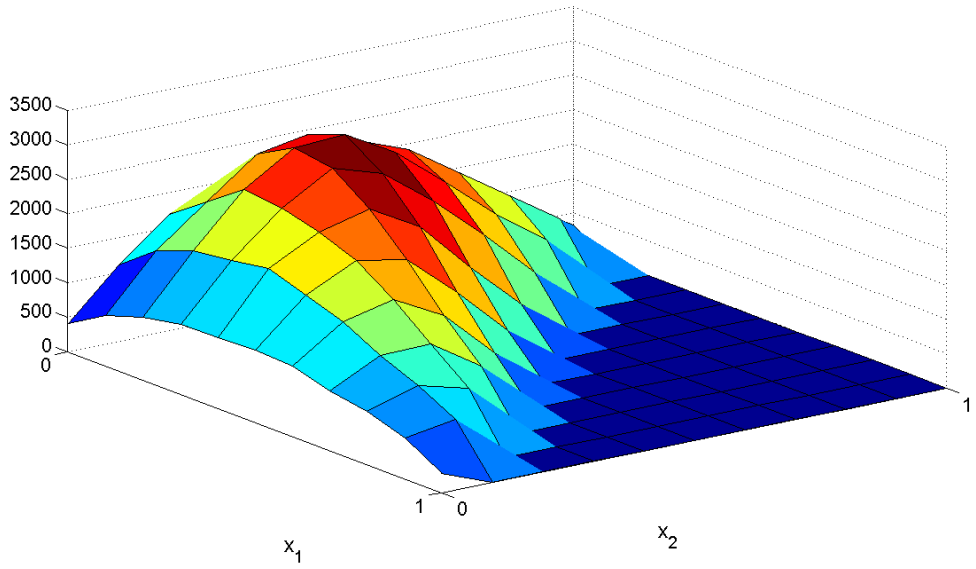

Figure 4.16: Simulation of the distribution with unknown density and $h = 1/10$, $T = 0.5$

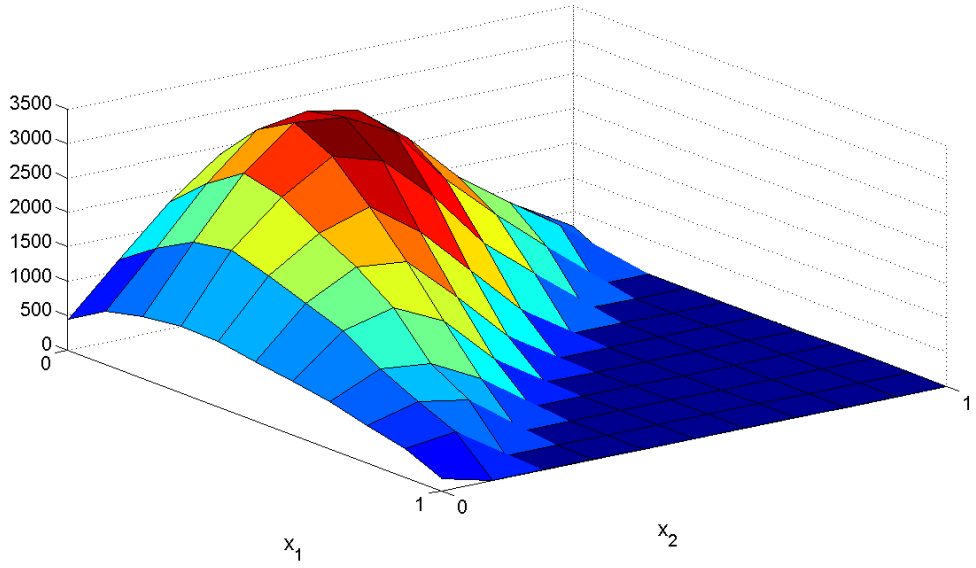

Figure 4.17: Simulation of the distribution with unknown density and $h = 1/10$, $T = 1$

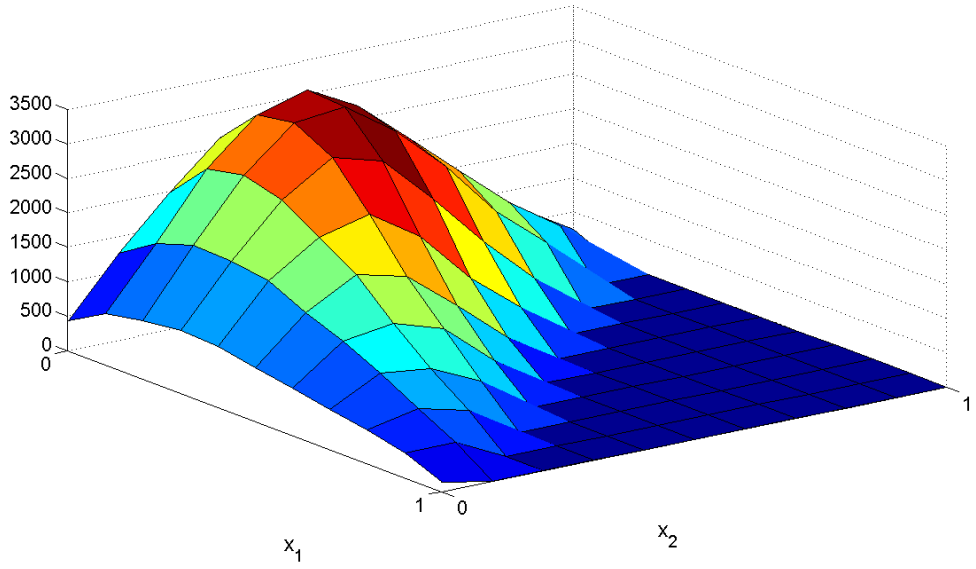

Figure 4.18: Simulation of the distribution with unknown density and $h = 1/10$, $T = 5$

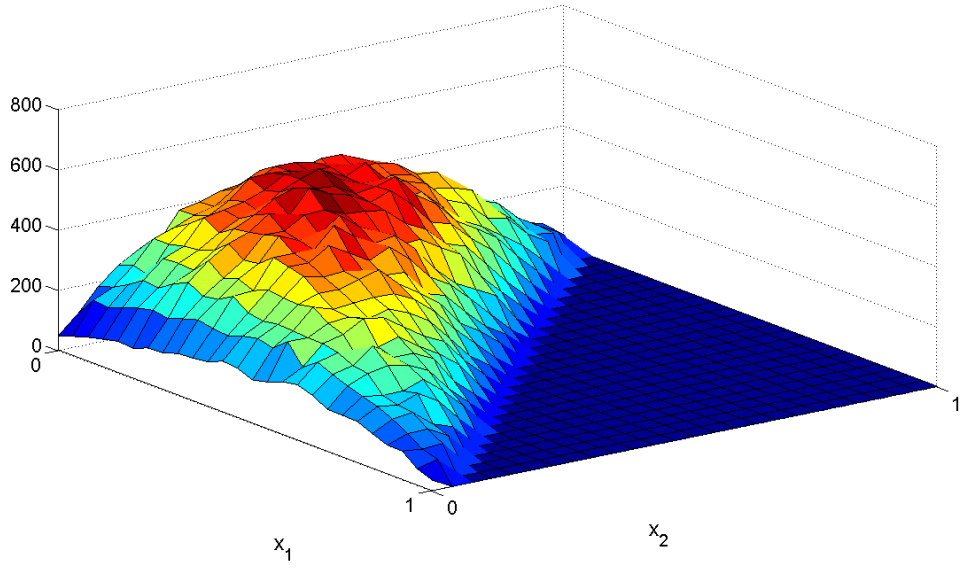

Figure 4.19: Simulation of the distribution with unknown density and $h = 1/25$, $T = 0.5$

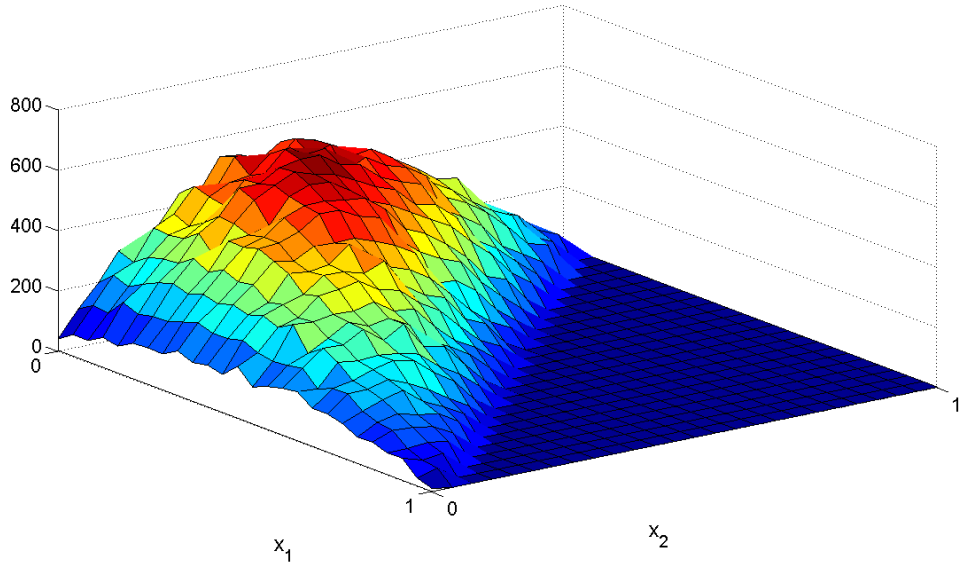

Figure 4.20: Simulation of the distribution with unknown density and $h = 1/25$, $T = 1$

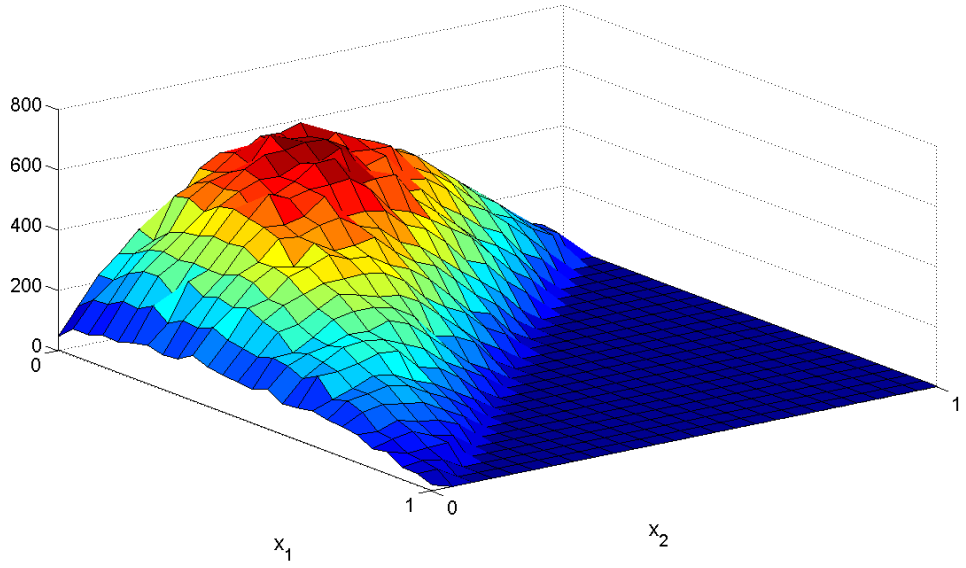

Figure 4.21: Simulation of the distribution with unknown density and $h = 1/25$, $T = 5$

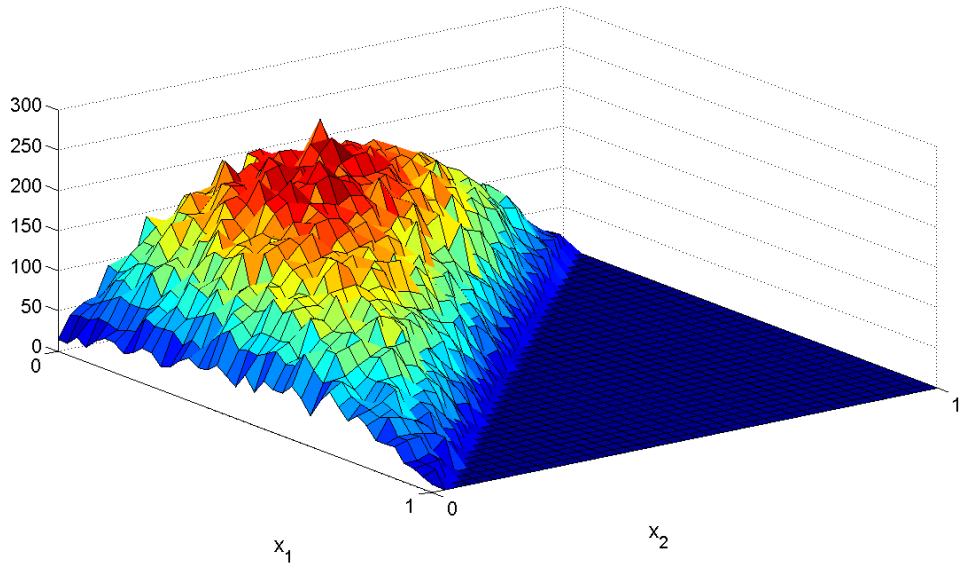

Figure 4.22: Simulation of the distribution with unknown density and $h = 1/40$, $T = 0.5$

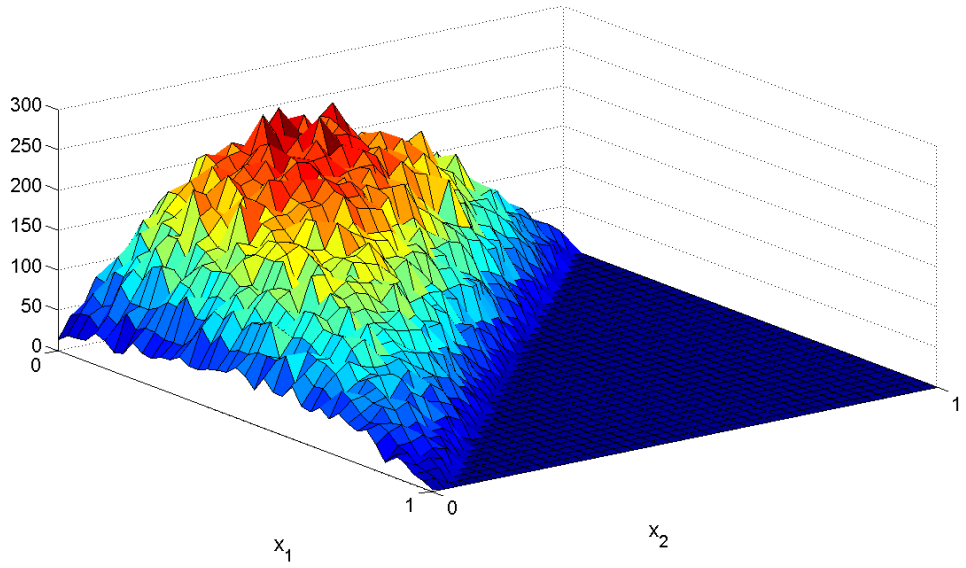

Figure 4.23: Simulation of the distribution with unknown density and $h = 1/40$, $T = 1$

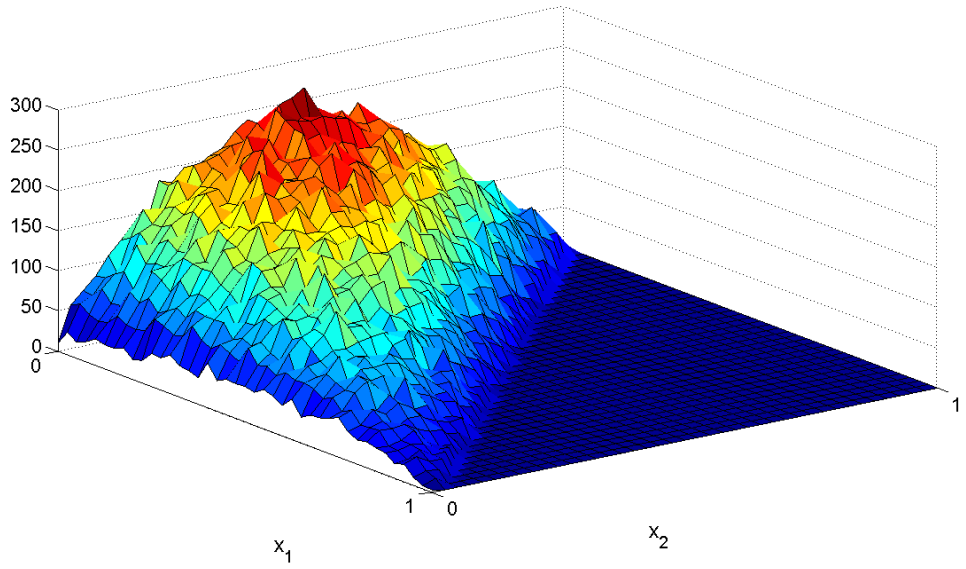

Figure 4.24: Simulation of the distribution with unknown density and $h = 1/40$, $T = 5$

Chapter 5

Conclusion

We investigate the evolution of proportions of alleles in a population over a long time horizon by simulating the stationary distribution of the Wright-Fisher diffusion using Markov Chain Monte Carlo techniques. The diffusion is approximated by a locally consistent, continuous time Markov chain. This approximation guarantees the weak convergence of the of the approximating process to the diffusion. We combine two discretization schemes for a locally consistent approximation. The first approach, a finite difference approach can not be used on all points on the diagonal. On these points, we use a direct, probabilistic approach to construct the required locally consistent Markov chain used for the simulation. We simulate distributions for choices of parameters for which analytical formulas for the densities are known to examine the behavior of the simulation. Finally, we simulate the stationary distribution of the diffusion for a choice of parameters for which no analytical formula for the stationary distribution is known.

Topics for further research include modifications of the approximating Markov chain, its state space and the dimensions of the model. The application of the direct, probabilistic approach not only on the diagonal, but on the whole state space can provide new approximating Markov chains, the convergence of different methods and a comparison with the algorithm we use, is of interest. Since the discretization schemes described here are developed for a very general setting, an application to a model with more than three alleles would be interesting. Furthermore, an

approximation on a non-uniform grid could be of advantage for distributions for which a lot of probability mass is concentrated in a certain region of the triangle.

Bibliography

[Br1999] Pierre Brmaud: Markov Chains, Gibbs Fields, Monte Carlo Simulation, and Queues, Springer-Verlag New York, Inc., New York (1999).

[KD2001] Harold J. Kushner, Paul Dupuis: Numerical Methods for Stochastic Control Problems in Continuous Time, Second Edition, Springer-Verlag New York Inc., New York (2001).

[EK1986] Stewart N. Ethier and Thomas G. Kurtz: Markov Processes: Characterization and Convergence, Wiley, New York (1986).

Appendix A

Code

```
function [ Xi_1,Xi_2,Xi_1_final,Xi_2_final,time ] =...
Wright_fisher(samplesize,T,m,mue)
time=zeros(samplesize,1);
for k=1: samplesize
    a=inf;
    b=inf;
    while a+b>m
        a=unidrnd(m+1)-1;
        b=unidrnd(m+1)-1;
    end
    Xi_1(k,1)=a/m;
    Xi_2(k,1)=b/m;
end
Xi_1_final=zeros(samplesize,1);
Xi_2_final=zeros(samplesize,1);
h=1/m;

%Initializing a
x=(0:h:1)';
a_1_1=repmat(x.*(1-x),1,m+1);
```

```
a_2_2=a_1_1';
a_1_2=-x*x';
a_2_1=a_1_2;

%Initializing b
b_1=-sum(mue(1,:))*repmat(x,1,m+1)+mue(1,1)*repmat(x,1,m+1)...
+mue(2,1)*repmat(x',m+1,1)+mue(3,1)*(1-repmat(x,1,m+1)-repmat(x',m+1,1));
b_2=-sum(mue(2,:))*repmat(x,1,m+1)'+mue(1,2)*repmat(x,1,m+1)...
+mue(2,2)*repmat(x',m+1,1)+mue(3,2)*(1-repmat(x,1,m+1)-repmat(x',m+1,1));

%Initializing Q
Q=a_1_1+a_2_2-0.5*(abs(a_1_2)+abs(a_2_1))+h*(abs(b_1)+abs(b_2));

%Initializing transition probabilities
p_1=(a_1_1/2-abs(a_1_2/2)+h*max(0,b_1))./Q;     %right
p_2=(a_2_2/2-abs(a_2_1/2)+h*max(0,b_2))./Q;     % up
p_3=max(0,-a_1_2)./(2*Q);                       %left up
p_4=(a_1_1/2-abs(a_1_2/2)+h*max(0,-b_1))./Q;    %left
p_5=(a_2_2/2-abs(a_2_1/2)+h*max(0,-b_2))./Q;    %down
p_6=max(0,-a_2_1)./(2*Q);                       %down right

dt=h^2./Q;

%Diagonal
for k1=1:length(x)
    k2=length(x)-k1+1;
    if b_2(k1,k2)>0
        Q(k1,k2)=-h*b_1(k1,k2)+x(k1)*x(k2);
        p_1(k1,k2)=0;
        p_2(k1,k2)=0;
```

```
        p_3(k1,k2)=(h*b_2(k1,k2)+x(k1)*x(k2)/2)/Q(k1,k2);
        p_4(k1,k2)=-(h*b_1(k1,k2)+h*b_2(k1,k2))/Q(k1,k2);
        p_5(k1,k2)=0;
        p_6(k1,k2)=x(k1)*x(k2)/(2*Q(k1,k2));
        dt(k1,k2)=h^2/Q(k1,k2);
    end

     if b_1(k1,k2)>0
        Q(k1,k2)=-h*b_2(k1,k2)+x(k1)*x(k2);
        p_1(k1,k2)=0;
        p_2(k1,k2)=0;
        p_3(k1,k2)=x(k1)*x(k2)/(2*Q(k1,k2));
        p_4(k1,k2)=0;
        p_5(k1,k2)=-(h*b_2(k1,k2)+h*b_1(k1,k2))/Q(k1,k2);
        p_6(k1,k2)=(h*b_1(k1,k2)+x(k1)*x(k2)/2)/Q(k1,k2);
        dt(k1,k2)=h^2/Q(k1,k2);
    end
end
%Simulation
for k=1:samplesize
     step=1;
     index1=round(m*Xi_1(k,step)+1);
     index2=round(m*Xi_2(k,step)+1);
     t=exprnd(dt(index1,index2));
     time(k,step)=t;
     while t<T
        index1=round(m*Xi_1(k,step)+1);
        index2=round(m*Xi_2(k,step)+1);
        z=rand();
```

```
prob1=p_1(index1,index2);
prob2=p_2(index1,index2);
prob3=p_3(index1,index2);
prob4=p_4(index1,index2);
prob5=p_5(index1,index2);
prob6=p_6(index1,index2);
if z<prob1
    Xi_1(k,step+1)=Xi_1(k,step)+h;
    Xi_2(k,step+1)=Xi_2(k,step);
else
    if z<prob1+prob2
        Xi_1(k,step+1)=Xi_1(k,step);
        Xi_2(k,step+1)=Xi_2(k,step)+h;
    else
        if z<prob1+prob2+prob3
            Xi_1(k,step+1)=Xi_1(k,step)-h;
            Xi_2(k,step+1)=Xi_2(k,step)+h;
        else
            if z<prob1+prob2+prob3+prob4
                Xi_1(k,step+1)=Xi_1(k,step)-h;
                Xi_2(k,step+1)=Xi_2(k,step);
            else
                if z<prob1+prob2+prob3+prob4+prob5
                    Xi_1(k,step+1)=Xi_1(k,step);
                    Xi_2(k,step+1)=Xi_2(k,step)-h;
                else
                    if z>=prob1+prob2+prob3+prob4+prob5
                        Xi_1(k,step+1)=Xi_1(k,step)+h;
                        Xi_2(k,step+1)=Xi_2(k,step)-h;
                    end
```

```
                        end

                    end
                end
            end
        end
        index1=round(m*Xi_1(k,step)+1);
        index2=round(m*Xi_2(k,step)+1);
        t=t+exprnd(dt(index1,index2));
        time(k,step)=t;
        Xi_1(k,step+1);
       Xi_2(k,step+1);
       Xi_1_final(k)=Xi_1(k,step+1);
       Xi_2_final(k)=Xi_2(k,step+1);
       step=step+1;
    end
end
end
```