
**The Moran model with recombination
and the long-term evolutionary
experiment with *E. coli* by R. E. Lenski**

Modelling, parameter estimation, and simulation

Dissertation

zur Erlangung des akademischen Grades
Doktor der Mathematik (Dr. math.)

vorgelegt an der Fakultät für Mathematik
der Universität Bielefeld

eingereicht von

Dipl.-Math. Sebastian Probst

am 10. September 2018

Betreuerin: Prof. Dr. E. Baake, Universität Bielefeld
Zweitgutachter: Prof. Dr. A. Wakolbinger, Goethe-Universität Frankfurt
Tag der mündlichen Prüfung: 25. Oktober 2018

Acknowledgements

I want to thank my supervisor Ellen Baake for giving me the opportunity to explore the field of mathematical biology and population genetics in particular, as well as for guiding me during my work for this thesis. She has always been accessible, have had confidence in me, and she was not becoming tired of encouraging me to stay persevering. Furthermore, I want to thank my colleagues and collaborators that have accompanied me on this journey. First of all, Anton Wakolbinger for asking me to join the very interesting research project about Lenski's long-term evolutionary experiment. Moreover, I am grateful to Marc Steinbach for the opportunity to use Clean and a special thank goes to his former PhD student Daniel Rose for many constructive discussions about the numerical approach in Chapter 3 and his proofreading. In addition, I like to thank all current and former members of the 'Biomathematics and Theoretical Bioinformatics' group for the pleasant and cooperative working atmosphere. In particular, Mareike Esser for the close cooperation that has led us to the main results of Chapter 2, as well as Sebastian Hummel and Fernando Cordero for proofreading this thesis.

Not least, I want to thank Gesine for her support and backing in the final stage of my doctorate.

Contents

Acknowledgements	i
List of Figures	v
List of Tables	vii
List of Algorithms	ix
1. Motivation and overview	1
2. The Moran model with recombination and the partitioning process	5
2.1. Introduction	5
2.2. Partitions and Möbius functions	8
2.3. The model and the genealogical approach	10
2.4. The partitioning process	16
2.5. Recombinators and sampling functions	25
2.6. Restrictions to subsystems	28
2.7. Duality	40
2.8. Applications and examples	47
2.9. Conclusion	50
3. Parameter estimation approach for the Moran model	51
3.1. Goal and outline	51
3.2. Moran model parameter estimation problem	52
3.3. NLP evaluation and details of implementation	54
3.4. Matrix representation of the generator and evaluation of the ODE system .	60
3.5. From simulation to observation data	67
3.6. Numerical experiments and results	71
3.7. Conclusion and prospect	81
4. Modelling and simulating Lenski's long-term evolutionary experiment	83
4.1. Introduction	83
4.2. A probabilistic model for the LTEE and its law of large numbers	88
4.3. Including clonal interference	97
4.4. Simulation algorithms	107
4.5. Discussion	109

Appendix	115
A. Constrained optimisation	115
A.1. Theory of constrained optimisation	115
A.2. Numerical algorithm for constrained optimisation	119
B. Multiple shooting approach for parameter estimation problems	127
B.1. Parameter estimation problem	127
B.2. Multiple shooting approach	128
C. Supplemental figures for Numerical results in Section 3.6	131
C.1. Results of experiments regarding different test cases	131
C.2. Time courses of the averaged normalised sampling functions	140
C.3. Runtimes of experiments regarding different test cases	144
Bibliography	147

List of Figures

2.1. Moran model with single crossover	6
2.2. Full ancestral recombination process	14
2.3. Marginalised ancestral recombination process	14
2.4. Construction of one possible ancestry fo a collection of sites	15
2.5. Marginal recombination probabilities	16
2.6. One step of the partitioning process	18
2.7. Correspondance of transitions forward and backward in time	21
2.8. Decomposition applied in the induction step of the proof of Lemma 3	33
2.9. Illustration for the proof of Proposition 4 in the case of four sites	35
3.1. Structure of the Jacobian of the continuity constraints	56
3.2. Structure of the Hessian of the Lagrangian function	57
3.3. Local update of the Hessian of the Lagrangian function	58
3.4. Flowchart of NLP-related evaluations	59
3.5. Sparse structure of the generator’s coefficient matrices for three sites	61
3.6. Concept of implementation for the generator	63
3.7. Comparison of memory requirement for different concepts of implementation	64
3.8. From simulations to observation data	69
3.9. Illustration of the hierarchy and iteration process to compute marginals . . .	71
3.10. Structure of the experimental environment	72
3.11. Schematic illustration of distribution shapes	73
3.12. Logarithmic observation grid and time courses of averaged sampling functions	75
4.1. Illustration of some day of Lenski’s LTEE and the subsequent sampling . . .	84
4.2. Empirical relative fitness data with error bars and corresponding power law	85
4.3. The relative fitness process and the approximating jump process	94
4.4. Cannings simulations (deterministic effects), disregarding clonal interference	96
4.5. Cannings simulations (deterministic effects), regarding clonal interference .	99
4.6. Heuristics simulations (deterministic effects), regarding clonal interference .	100
4.7. Cannings simulations (effects following exponential distribution)	105
4.8. Heuristics simulations (effects following exponential distribution)	105
4.9. Cannings simulations (effects following shifted Pareto distribution)	106
4.10. Heuristics simulations (effects following shifted Pareto distribution)	107
4.11. Synchronous growth model with equally-sized clones at the end of the day .	110

A.1. Showcase for a constrained nonlinear optimisation problem	116
A.2. Filter concept in constrained optimisation	123
A.3. Second order correction remedying the Maratos effect	124
B.1. Multiple shooting approach	129
C.1. Results of a subset of all experiments regarding [8000:uniform:6;1.0;full]	131
C.2. Results of all experiments regarding [8000:uniform:6;1.0;full]	132
C.3. Results of all experiments regarding [1600:uniform:6;1.0;full]	133
C.4. Results of all experiments regarding [8000:logarithmic:6;1.0;full]	134
C.5. Results of all experiments regarding [1600:logarithmic:6;1.0;full]	135
C.6. Results of all experiments regarding [1600:logarithmic:16;1.0;full]	136
C.7. Results of all experiments regarding [1600:logarithmic:16;0.1;full]	137
C.8. Results of all experiments regarding [1600:logarithmic:16;0.1;reduced]	138
C.9. Change of exit status regarding [●:logarithmic:6;1.0;full]	139
C.10.Histograms of the exit statuses' relative frequencies for all test cases	139
C.11.Time courses of normalised sampling functions for two sites	140
C.12.Time courses of normalised sampling functions for three sites	141
C.13.Time courses of normalised sampling functions for four sites	142
C.14.Time courses of normalised sampling functions for five sites	143
C.15.Average SQP/NLP CPU times regarding [1600:logarithmic:●;1.0;full]	144
C.16.Average SQP/NLP CPU times regarding [1600:logarithmic:16;0.1;●]	145
C.17.Total SQP solution times for each experiment	146

List of Tables

2.1.	Total number and ratio of non-zero transitions of the partitioning process	20
2.2.	Transitions of the partitioning process connected with the previous approach	22
3.1.	Fixed settings for the numerical experiments	73
3.2.	Summary of parameter values determining the numerical experiments	76
3.3.	Number of non-zero elements for crucial NLP components (up to six sites)	79
3.4.	Saving of arithmetic operations for reduced accuracy (five, six sites)	80
4.1.	Summary of parameter estimates and corresponding simulation results	114
A.1.	Algorithmic constants of a filter line-search SQP algorithm	124

List of Algorithms

1.	NLP evaluation cascade	60
2.	Constructor of the generator's structure	65
3.	Update the generator's data	66
4.	Linear map defined by the generator	67
5.	Iterative calculation of all marginals	70
6.	Simulating Lenski's experiment (Cannings model)	108
7.	Simulating Lenski's experiment (thinning heuristics)	109
8.	Basic Newton method for equality constrained NLP	120
9.	Filter line-search SQP algorithm	125

Chapter 1.

Motivation and overview

The field of population genetics comprises mathematical models of genetical variation between and within populations and was established in the first half of the 20th century by the works of its founding fathers Ronald Fisher, J.B.S. Haldane and Sewall Wright (in alphabetical order of their surnames). These models deal with the change (or evolution) of existing variants (alleles) of one or more genes, within one or between several idealised (sub-) populations of constant or fluctuating size; see, for example, [31] for a historical and theoretical introduction. Broadly summarised, there are five main processes that drive these models: *genetic drift*, *recombination*, and *mutation* on the level of gametes, as well as *migration* and *natural selection* on the level of individuals' phenotypes.

Genetic drift means neutral (i.e. independent of the alleles) random change of allele frequencies between generations, see [69] for a propaedeutic discussion. The most basic (reference-) models for one gene with two alleles under genetic drift are the *Wright-Fisher* and the *Moran* model. Both assume constant, finite populations and can be described as Markov processes. However, they model reproduction schemes differently. The first one (named after Sewall Wright and Ronald Fisher) uses synchronous and discrete generations and represents drift by randomly sampling the alleles with replacement from the previous generation. The second one (named after Patrick Moran) permits overlapping generation in continuous time. In each moment of reproduction, one individual is chosen to reproduce and its single offspring replaces one individual that has to die. A third model class called *Cannings models* (named after Chris Cannings, 1974) generalises the reproduction scheme of the Wright-Fisher approach by allowing each individual to give birth to a random number of offspring. This random number has the same distribution among individuals, and individuals reproduce independently of each other.

Recombination denotes the partial exchange of genetic sequences between two gametes (as it happens during sexual reproduction). Mutations are random changes of alleles that may be caused by errors during replication or several types of environmental influences. They can be categorised as beneficial, neutral, and deleterious, depending on their influence on the *fitness* of the individual, i.e. how likely it is to survive and how successful it is at reproducing. The key mechanism described by selection is the differential survival and reproductive success of individuals due to their different phenotypes.

In this thesis we deal with extensions of two of the above-mentioned models for genetic drift, namely a multilocus version of the Moran model with recombination and a Cannings model with mutation and selection. A model treating migration is beyond the scope of this thesis. For both models, the first main goal is to develop previous works ([16], [17], [42], [97], [47]) further both conceptually and mathematically. The second goal is to provide for possibilities to estimate model parameters and the third goal is to validate the findings numerically.

In addition to the dedicated and detailed introductions and outlines at the beginning of each chapter, let us give a brief preview. In Chapter 2, first, the Moran population process (forward in time) is recapitulated. Then, a partition-valued Markov process (backward in time) is introduced, which experiences splitting and coalescence, and turns out to be dual to the Moran population process with respect to a specific sampling function. This sheds new light on the work in [16], [17]. Several properties of the sampling function and the partition-valued Markov process (especially due to restriction to subsystems) as well as limiting results are stated. One of these findings, an ordinary differential equation for the expectations of the sampling function, allows to formulate a general approach to estimate the inherent recombination distribution by using methods from nonlinear programming in Chapter 3. We not only state the numerical model as a constrained nonlinear optimisation problem and go into computational details, but also propose an efficient way for its implementation. Finally, for this part of the thesis, a test environment is created and evaluated to give empirical assessment on the influence of the quality of the observation data on the model fit.

In Chapter 4, we turn to a famous, ongoing wet lab experiment known as *Lenski's long-term evolutionary experiment*. First, we describe the experiment with the help of a Cannings model with mutation and selection, which provides insight into the growth dynamics within every single day and reveals a runtime effect. After the review of the 'dynamical law of large numbers' result in [47], we return to finite populations and develop approximations and heuristics, among them a refinement and extension of the one in [42] for the case of competing beneficial mutations. Those heuristics allow to estimate model parameters directly from the available experimental data. We support the finding with simulations of the Cannings model and our heuristics using those estimates. Concluding, we compare and discuss our results with previous ones by Lenski and his collaborators in [97].

Wide parts of this thesis are results of collaborations and have already been published or submitted to scientific journals. Most of Chapter 2 arose from a joint project with (at that time PhD student) Dr. Mareike Esser together with our common supervisor Prof. Dr. Ellen Baake. The questions to be tackled were separate initially, but grew together more and more throughout the project, so that the contribution of Dr. Esser and the author of this thesis can not be disentangled in detail. Consequently, both theses contain the results of the common publication [29] and complement them with independent additional findings.

Specifically, here, Section 2.4 is extended by Corollary 1 up to the paragraph about the connection with previous work by Bobrowski et al.; the latter is itself an expansion of Remark 3 in [29]. Furthermore, the larger part of Section 2.6, beginning after Remark 6, expands the restriction to subsystems to the normalised sampling functions and the partitioning process, which are necessary to prove the marginalisation consistency of the duality result in Section 2.7. The verification of this property was still missing in [29].

Chapter 4 is the result of a common project with Prof. Dr. Ellen Baake, Prof. Dr. Anton Wakolbinger, and Dr. Adrián González Casanova, which has been submitted as [4] just prior to the submission of this thesis.

Chapter 2.

The Moran model with recombination and the partitioning process

2.1. Introduction

Models that describe the evolution of finite populations under recombination are among the major challenges in population genetics. This chapter is devoted to the *Moran model with recombination* (in continuous time), which is briefly described as follows (see [26, 17]).

Preliminary description of the model and two main lines of research. A chromosome is identified with a linear arrangement (or *sequence*) of n discrete positions called *sites*, which are collected in the set $S = \{1, \dots, n\}$. A site may be understood as a nucleotide site or a gene locus. We will throughout consider chromosomes as (haploid) *individuals*, that is, we think at the level of gametes (rather than that of diploid individuals that carry two copies of the genetic information). Site i is occupied by letter $x_i \in \mathbb{X}_i$, where \mathbb{X}_i is a finite set, $1 \leq i \leq n$. If sites are nucleotide sites, a natural choice for each \mathbb{X}_i is the nucleotide alphabet $\{A, G, C, T\}$; if sites are gene loci, \mathbb{X}_i is the set of alleles that can occur at locus i . The genetic type of each individual is thus described by the sequence $x = (x_1, x_2, \dots, x_n) \in \mathbb{X}_1 \times \dots \times \mathbb{X}_n =: \mathbb{X}$, where \mathbb{X} is the type space. Recombination means that a new individual is formed as a ‘mixture’ of an (ordered) pair of parents, say x and y . We will restrict ourselves to *single-crossover recombination*, that is, the offspring inherits the leading segment (up to site i , for some $1 \leq i < n$) from the first and the trailing segment (after site i) from the second parent. The recombined type thus is $(x_{\leq i}, y_{> i}) := (x_1, \dots, x_i, y_{i+1}, \dots, y_n)$; we say that a crossover has happened between sites i and $i + 1$. The sites that come from the paternal and the maternal sequence, respectively, define a *partition* \mathcal{A} of S into two parts (we need not keep track of which part was ‘maternal’ and which was ‘paternal’). All partitions of S into two ordered (or contiguous) parts ($\mathcal{A} = \{\{1, 2, \dots, i\}, \{i + 1, \dots, n\}\}, i \in S \setminus \{n\}$) can be realised, via a single crossover event. Altogether, whenever an offspring is created, its sites are partitioned between parents according to \mathcal{A} with probability $r_{\mathcal{A}}$, where $r_{\mathcal{A}} \geq 0$, $\sum_{\mathcal{A} \in \mathbb{O}_2(S)} r_{\mathcal{A}} \leq 1$, and $\mathbb{O}_2(S)$ is the set of all ordered partitions of S into two parts. Let us note that, due to the one-to-one correspondence between elements of $S \setminus \{n\}$ and those of $\mathbb{O}_2(S)$, the specification of the

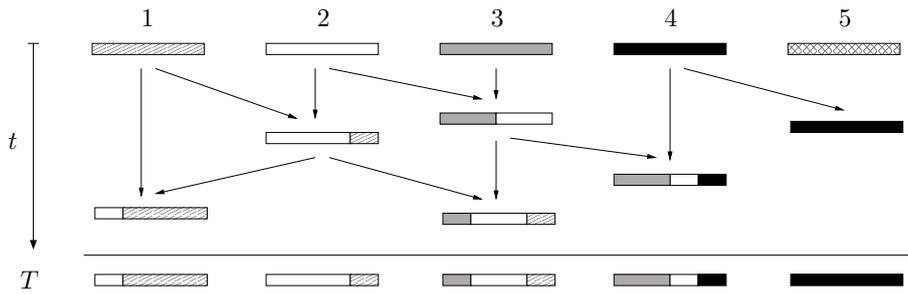


Figure 2.1.: Snapshot of a Moran model realisation with $N = 5$ individuals. For example, in the first event, individual 3 dies and is replaced by a recombined copy of individuals 2 and 3. The last line shows the composition of the population at the final timepoint, T .

$r_{\mathcal{A}}$ simply means that a crossover probability is associated with each site in $S \setminus \{n\}$. The sum $\sum_{\mathcal{A} \in \mathbb{O}_2(S)} r_{\mathcal{A}}$ is the probability that some recombination event takes place during reproduction. With probability $r_{\{S\}} = 1 - \sum_{\mathcal{A} \in \mathbb{O}_2(S)} r_{\mathcal{A}}$, there is no recombination, in which case the offspring is the full copy of a single parent. We write $\mathbb{O}_{\leq 2}(S) := \mathbb{O}_2(S) \cup \{S\}$ for the set of ordered partitions into at most two parts. The collection $\{r_{\mathcal{A}}\}_{\mathcal{A} \in \mathbb{O}_{\leq 2}(S)}$ is known as the *recombination distribution* (see [21, p. 55]).

Consider now a *population* of a constant number of N haploid individuals (that is, gametes), which evolves as follows (see Figure 2.1). Each individual has an exponential lifespan with parameter 1 (this choice of the parameter is without loss of generality; it simply sets the time scale). When an individual dies, it is replaced by a new one as follows. First draw a partition \mathcal{A} according to the recombination distribution. Then draw $|\mathcal{A}|$ parents from the population (the parents may include the individual that is about to die), uniformly and with replacement, where $|\mathcal{A}|$ is the number of parts in \mathcal{A} . If $|\mathcal{A}| = 2$, the offspring inherits the leading segment of \mathcal{A} from the first and the trailing segment from the second parent, as described above. If $|\mathcal{A}| = 1$ (and thus $\mathcal{A} = \{S\}$), the offspring is a full copy of a single parent (again chosen uniformly from among all individuals); this is called a (*pure*) *resampling* event. All events are independent of each other.

Note that it may seem biologically more realistic to draw two parents *without* replacement. However, assuming sampling *with* replacement entails significant simplifications, and yields the same process as sampling without replacement with a slight change in the recombination distribution. More precisely, since drawing the same individual twice means that the offspring is a full copy of this single parent, our process agrees (in distribution) with the analogous process without replacement if $r_{\mathcal{A}}$ is replaced by $r_{\mathcal{A}}(N-1)/N$ for all $\mathcal{A} \in \mathbb{O}_2(S)$ (and $r_{\{S\}}$ is set accordingly). The model will be described more formally later.

For now, let us summarise the two main lines of research in this context. On the one hand, there has been considerable interest in how the composition of the population evolves over time, and, in particular, how the correlations between sites (known as linkage disequilibria) will develop; see the overviews in [56, Chap. 5.4], [26, Chap. 3.3 and 8.2], or

[94, Chap. 7.2.4]. Since there is no mutation, a single type will go to fixation in the long run, that is, the entire population will ultimately consist of this single type. In the absence of recombination, this will be one of the types initially present, and it is well known that the fixation probability for a given type equals its initial frequency. If there is recombination, the type that ultimately wins can also be a newly-composed type, but little is known about the fixation probabilities of the many possible types. The explicit development over time is even more challenging, due to an intricate interplay of resampling and recombination. It is usually approached forward in time, e.g., [74], [79], [80], [86], [5], [26, Chap. 8.2], or [17]. In the deterministic limit, which emerges when $N \rightarrow \infty$ without rescaling of the $r_{\mathcal{A}}$ or of time, the population is described by a system of ordinary differential equations, again forward in time. This system has an explicit solution, both for the type distribution and for correlation functions of all orders, for an arbitrary number of sites (see [9] and [8]). This also provides a decent approximation for large but finite populations in [5], but dealing appropriately with the stochasticity of finite populations remains a major challenge.

The second line of research is concerned with *genealogical aspects* and sampling formulae. Here, one starts from a sample taken from the present population and traces back the ancestry of the various segments the individuals are composed of. A major challenge lies in the calculation of the probabilities for the type distribution of a random sample, that is, one aims at so-called *sampling formulae*, see [26, Chap. 3.6]. These questions are naturally approached backward in time. Usually, one employs the *diffusion (or weak recombination) limit*, that is, time is sped up by a factor of N , followed by $N \rightarrow \infty$ such that $Nr_{\mathcal{A}} \rightarrow \varrho_{\mathcal{A}}$, $\varrho_{\mathcal{A}}$ a constant, $\mathcal{A} \in \mathbb{O}_2(S)$. Obtaining sampling formulae is tied to the situation in which the population has reached a stationary state; even this case is very hard to treat, and coping with time dependence seems to be hopeless.

Goal and outline. The goal of this chapter is to build a bridge between these two lines of research. We will explore the type distribution and the correlations over time, in the stochastic setting. A starting point will be a recent paper [17], whose authors approach this question. Their setting is entirely forward in time, which effectively hides some of the underlying structure. In contrast, we will proceed backward in time and provide a genealogical approach for the analysis of correlations. The crucial notion in this context will be that of *duality* between the original Moran model forward in time and a suitable ancestral process that follows back the ancestry of selected segments from today's population. This will also shed new light on the results of [17]. In order to keep the approach as general as possible, we will throughout adhere to the original (finite N) model, without taking any limit, but will discuss the various scalings and limits where appropriate.

The chapter is organised as follows. In Section 2.2, we start by collecting some important facts about partitions and Möbius functions. We then (Section 2.3) introduce the model more formally and motivate our genealogical approach, which may be considered a marginal

version of the usual ancestral process with recombination. In Section 2.4, we describe our ancestral process, which is a partitioning process that keeps track of how the ancestral material is partitioned between individuals and connect it to the previous work in [17]. In Section 2.5, we introduce a systematic description via recombinators, which describe the action of recombination on a population and have proved very useful in the deterministic setting. We complement them here by *sampling functions*, which are additionally required for finite populations. In Section 2.6 we study the impact of restriction to subsystems on the recombinators, correlations as well as sampling functions and show that the partitioning process is lumpable. The collection of sampling functions will be crucial since it will also serve as *duality function* in Section 2.7, where the duality between the Moran model forward in time and the partitioning process backward in time is proved. This proof, at the same time, yields a differential equation system for the expectations of the sampling functions, which are the building blocks for the linkage disequilibria. In Section 2.8, we apply our results to the cases of two and three sites. We will see that the expected linkage disequilibria (of second and third order) decay exponentially even in the presence of resampling, and identify further linear combinations of expected sampling functions that decay exponentially. For two sites, we also obtain the explicit time course for the expected composition of the population, and, at the same time, the fixation probabilities of the various types.

2.2. Partitions and Möbius functions

Working with partitions will be essential to our approach, and we will rely throughout on the powerful concept of *Möbius functions* and *Möbius inversion*. Let us briefly collect the basic notions and standard results; more background material as well as the proofs may be found in [83], [13, Chap. 3.2],[1, Chap. I,II,IV] and [87, Chap. 3].

Partitions. Let W be a finite, nonempty, totally ordered set, such as a finite subset of \mathbb{N} ; later, W will be S or a subset thereof. Let $\mathbb{P} = \mathbb{P}(W)$ be the set of partitions of W . We write such a partition as $\mathcal{A} = \{A_1, \dots, A_m\}$, where $A_j \neq \emptyset$ for all j and $A_j \cap A_k = \emptyset$ for all $j \neq k$ together with $A_1 \cup \dots \cup A_m = W$. We call A_j a *block (or part)* of \mathcal{A} and $m = |\mathcal{A}|$ is the number of blocks in \mathcal{A} .

We say that a partition $\mathcal{A} = \{A_1, \dots, A_m\}$ of W is *ordered (or contiguous, or an interval partition)* if every A_j is ordered in W , that is, $A_j = \{x \in W \mid \min A_j \leq x \leq \max A_j\}$. For example, if $W = \{1, 2, 5, 7, 9\}$, then $\{\{1, 2, 5\}, \{7, 9\}\}$ is ordered, but $\{\{1, 2, 7\}, \{5, 9\}\}$ is not. The set of all ordered partitions of W is denoted by $\mathbb{O}(W)$, the set of all ordered partitions of W into (exactly) two parts is $\mathbb{O}_2(W)$, and the set of all ordered partitions of W into at most two parts is $\mathbb{O}_{\leq 2}(W)$.

For a given partition $\mathcal{A} = \{A_1, \dots, A_m\}$ of W , let $M := \{1, 2, \dots, m\} = M(\mathcal{A})$ and, for $J \subseteq M$, we define $\mathcal{A}_J := \{A_j\}_{j \in J}$ and $A_J := \cup_{j \in J} A_j$. \mathcal{A}_J is a partition of A_J . In particular,

$\mathcal{A}_M = \mathcal{A}$, $A_M = W$, $\mathcal{A}_{\{j\}} = \{A_j\}$, and $\mathcal{A}_{M \setminus \{j\}} = \mathcal{A} \setminus \{A_j\}$, for any $j \in M$. Note that M depends on \mathcal{A} , but we suppress this dependence when there is no risk of confusion. We will throughout abbreviate $J \setminus j := J \setminus \{j\}$ and $J \cup k := J \cup \{k\}$.

The natural ordering relation on $\mathbb{P}(W)$ is denoted by \preceq , where $\mathcal{A} \preceq \mathcal{B}$ means that \mathcal{A} is a *refinement* of \mathcal{B} , that is, every block of \mathcal{A} is a subset of a block of \mathcal{B} ; equivalently, \mathcal{B} is a *coarsening* of \mathcal{A} . $\mathcal{A} \prec \mathcal{B}$ means that $\mathcal{A} \preceq \mathcal{B}$ and $\mathcal{A} \neq \mathcal{B}$. Together with the resulting partial order, $\mathbb{P}(W)$ is a *poset* and, in particular, a *finite lattice*. $\mathbb{P}(W)$ has a unique *minimal* or *finest* partition, which is denoted as $\mathbf{0} = \{\{x\} \mid x \in W\}$; likewise, there is a unique *maximal* or *coarsest* one, namely $\mathbf{1} = \{W\}$.

When U and V are *disjoint* (finite) sets, two partitions $\mathcal{A} \in \mathbb{P}(U)$ and $\mathcal{B} \in \mathbb{P}(V)$ can be joined into $\mathcal{A} \cup \mathcal{B}$ to form an element of $\mathbb{P}(U \dot{\cup} V)$. Furthermore, if $U \subseteq W$, a partition $\mathcal{A} \in \mathbb{P}(W)$, with $\mathcal{A} = \{A_1, \dots, A_m\}$ say, defines a unique partition of U by restriction. The latter is denoted by $\mathcal{A}|_U$, and its parts are precisely all non-empty sets of the form $A_i \cap U$ with $1 \leq i \leq m$. In particular, $\mathbf{1}|_U$ is the coarsest element in $\mathbb{P}(U)$. For two partitions \mathcal{A} and \mathcal{B} , the *least upper bound* will be denoted by $\mathcal{A} \vee \mathcal{B}$, namely the finest partition \mathcal{C} for which $\mathcal{A} \preceq \mathcal{C}$ and $\mathcal{B} \preceq \mathcal{C}$. Analogously define the *greatest lower bound* of \mathcal{A} and \mathcal{B} by $\mathcal{A} \wedge \mathcal{B}$.

Example 1. Consider $W = \{1, \dots, 5\}$ and the two partitions $\mathcal{A} = \{\{1, 3, 4\}, \{2, 5\}\}$ and $\mathcal{B} = \{\{1, 4\}, \{2, 3\}, \{5\}\}$ thereof together with a subset $U = \{1, 2, 4\}$ of W . Then $\mathcal{A} \wedge \mathcal{B} = \{\{1, 4\}, \{2\}, \{3\}, \{5\}\}$, $\mathcal{A} \vee \mathcal{B} = \{\{1, \dots, 5\}\}$, and $\mathcal{A}|_U = \{\{1, 4\}, \{2\}\}$.

Möbius functions on the poset of partitions and Möbius inversion. The *Möbius function* of a poset is a general and powerful tool in discrete mathematics. It may be considered as a systematic way of implementing the inclusion-exclusion principle. We rely on it in two contexts here: First, we use it to turn sampling without replacement into sampling with replacement, and vice versa. Second, we need it to turn type frequencies into linkage disequilibria.

Referring to [1, Prop. 4.6], let us only summarise here that the Möbius function μ is defined for all $\mathcal{A} \preceq \mathcal{C} \in \mathbb{P}(W)$ via

$$\sum_{\mathcal{A} \preceq \mathcal{B} \preceq \mathcal{C}} \mu(\mathcal{A}, \mathcal{B}) = \begin{cases} 1, & \mathcal{A} = \mathcal{C}, \\ 0, & \text{otherwise,} \end{cases} \quad (2.1)$$

where the underdot indicates the summation variable. Let $\mathcal{A} \preceq \mathcal{B} \in \mathbb{P}(W)$, with $m = |\mathcal{B}|$ the number of blocks in \mathcal{B} , and n_j the number of blocks of \mathcal{A} within block B_j of \mathcal{B} , that is, n_j is the number of blocks in $\mathcal{A}|_{B_j}$, $1 \leq j \leq m$. The Möbius function of the pair $(\mathcal{A}, \mathcal{B})$ is then given by

$$\mu(\mathcal{A}, \mathcal{B}) = \prod_{j=1}^m \mu(\mathcal{A}|_{B_j}, \mathbf{1}|_{B_j}) = \prod_{j=1}^m (-1)^{n_j-1} (n_j - 1)!, \quad (2.2)$$

see [83, Sect. 7, Ex. 1] or [13, Chap. 3.2, Ex. 4]. We can now state the fundamental Möbius inversion principle as in [1, Prop. 4.18]. Let f and g be mappings from $\mathbb{P}(W)$ to \mathbb{C} which are, for all $\mathcal{A} \in \mathbb{P}(W)$, related via

$$g(\mathcal{A}) = \sum_{\mathcal{B} \succ \mathcal{A}} f(\mathcal{B}). \quad (2.3)$$

Then, this can be solved for f via the inversion formula

$$f(\mathcal{A}) = \sum_{\mathcal{B} \succ \mathcal{A}} \mu(\mathcal{A}, \mathcal{B}) g(\mathcal{B}). \quad (2.4)$$

More precisely, this is *inversion from above*. An analogous formula applies for *inversion from below*; this relies on refinements rather than coarsenings, with ‘ \succ ’ replaced by ‘ \preceq ’ in (2.3) and (2.4). It is important to note that Möbius inversion is not restricted to functions; it also applies to bounded operators.

2.3. The model and the genealogical approach

In this section, we define the model formally and motivate our genealogical approach.

The Moran model with single-crossover recombination. We identify the population at time t by a (random) counting measure Z_t on \mathbb{X} . Namely, $Z_t(\{x\})$ denotes the number of individuals of type $x \in \mathbb{X}$ at time t , and $Z_t(\mathbb{A}) := \sum_{x \in \mathbb{A}} Z_t(\{x\})$ for $\mathbb{A} \subseteq \mathbb{X}$; we abbreviate $Z_t(\{x\})$ as $Z_t(x)$. If we define δ_x as the point measure on x (i.e., $\delta_x(y) = \delta_{x,y}$ for $x, y \in \mathbb{X}$), we can also write $Z_t = \sum_{x \in \mathbb{X}} Z_t(x) \delta_x$. Since our Moran population has constant size N , we have $\|Z_t\| = N$ for all times, where $\|Z_t\| := \sum_{x \in \mathbb{X}} Z_t(x) = Z_t(\mathbb{X})$ is the norm (or total variation) of Z_t .

So, $\{Z_t\}_{t \geq 0}$ is a Markov process in continuous time with values in

$$E := \{z \in \{0, \dots, N\}^{|\mathbb{X}|} \mid \|z\| = N\}, \quad (2.5)$$

where $|\mathbb{X}|$ is the number of elements in \mathbb{X} . We will describe the action of recombination on (positive) measures with the help of so-called *recombinators* as introduced in [9]; see also [5] for a pedestrian introduction. Let $\mathcal{M}_+(\mathbb{X})$ be the set of all positive, finite measures on \mathbb{X} and we understand $\mathcal{M}_+(\mathbb{X})$ to include the zero measure. Define the canonical projection $\pi_I: \mathbb{X} \rightarrow \times_{i \in I} \mathbb{X}_i =: \mathbb{X}_I$, for $I \subseteq S = \{1, \dots, n\}$, by $\pi_I(x) = (x_i)_{i \in I}$ as usual. For $\omega \in \mathcal{M}_+(\mathbb{X})$, the shorthand $\pi_I \cdot \omega := \omega \circ \pi_I^{-1}$ indicates the marginal measure with respect to the sites in $I \subseteq S$, where π_I^{-1} is the preimage of π_I . The operation \cdot (where the dot is on the line and should not be confused with a multiplication sign) is known as the

pushforward of ω w.r.t. π_I . In terms of coordinates, the definition may be spelled out as

$$(\pi_I.\omega)(x_I) = \omega \circ \pi_I^{-1}(x_I) = \omega(\{x \in \mathbb{X} \mid \pi_I(x) = x_I\}), \quad x_I \in \mathbb{X}_I.$$

Note that $\pi_\emptyset.\omega = \|\omega\|$ and $\pi_S.\omega = \omega$.

Consider now $\mathcal{A} = \{\{1, 2, \dots, i\}, \{i+1, \dots, n\}\} \in \mathbb{O}_2(S)$ and $\omega \in \mathcal{M}_+(\mathbb{X}) \setminus 0$, and define the *projective recombinator* as

$$R_{\mathcal{A}}^p(\omega) := \frac{1}{\|\omega\|^2} (\pi_{\{1, \dots, i\}}.\omega) \otimes (\pi_{\{i+1, \dots, n\}}.\omega), \quad (2.6)$$

where \otimes indicates the tensor product (or product measure). Moreover, we set $R_{\mathcal{A}}^p(\omega) := \omega/\|\omega\|$. $R_{\mathcal{A}}^p(\omega)$ is a probability measure for *all* $\omega \in \mathcal{M}_+(\mathbb{X}) \setminus 0$, where the zero measure is excluded to make it well-defined. In words, $R_{\mathcal{A}}^p$ turns ω into the (normalised) product measure of its marginals with respect to the blocks in \mathcal{A} . Writing out (2.6) in terms of coordinates gives

$$\begin{aligned} (R_{\mathcal{A}}^p(\omega))(x) &= \frac{1}{\|\omega\|^2} (\pi_{\{1, \dots, i\}}.\omega)(x_{\{1, \dots, i\}}) (\pi_{\{i+1, \dots, n\}}.\omega)(x_{\{i+1, \dots, n\}}) \\ &= \frac{1}{\|\omega\|^2} \omega(x_1, \dots, x_i, *, \dots, *) \omega(*, \dots, *, x_{i+1}, \dots, x_n), \end{aligned}$$

where $*$ means marginalisation. If $\omega = z$ is the current population, then $R_{\mathcal{A}}^p(z)$ is the type distribution that results when a new individual is created by drawing a leading and (possibly) a trailing segment (as encoded by $\mathcal{A} \in \mathbb{O}_{\leq 2}(S)$) from the current population, uniformly and with replacement.

Remark 1. $R_{\mathcal{A}}^p$ is a projective version of the recombinator defined in [9]; it differs from the latter by a factor of $1/\|\omega\|$. Clearly, both versions agree on the set of probability measures. As we shall see, the projective version is more suitable in the stochastic setting, while the original recombinators are better adapted to the deterministic situation. Since recombinators will only appear in the projective version in this chapter, we will drop the superscript and the specification ‘projective’ and call $R_{\mathcal{A}} := R_{\mathcal{A}}^p$ a recombinator by slight abuse of language.

In Section 2.5, we will generalise the recombinators and learn more about their probabilistic meaning and mathematical properties. For the moment, let us use them to reformulate the Moran model with recombination in a compact way. Namely, since all individuals die at rate 1, the population loses type- y individuals at rate $Z_t(y)$. Each loss is replaced by a new individual, which is sampled uniformly from $R_{\mathcal{A}}(Z_t)$ with probability $r_{\mathcal{A}}$, $\mathcal{A} \in \mathbb{O}_{\leq 2}(S)$. Therefore, when $Z_t = z$, the transition to $z + \delta_x - \delta_y$ occurs with rate

$$\lambda(z; y, x) := \sum_{\mathcal{A} \in \mathbb{O}_{\leq 2}(S)} r_{\mathcal{A}} (R_{\mathcal{A}}(z))(x) z(y). \quad (2.7)$$

The summand for $\mathcal{A} = \mathbf{1}$ corresponds to pure resampling, whereas all other summands include recombination. Note that λ includes ‘silent transitions’ ($x = y$).

Remark 2. *We would like to mention that the model may alternatively be formulated in terms of reproducing individuals rather than dying individuals, as follows. Each individual reproduces at rate 1 and picks an $\mathcal{A} \in \mathbb{O}_{\leq 2}(S)$ according to the recombination distribution. If $\mathcal{A} \in \mathbb{O}_2$, the reproducing individual contributes the sites in one of the blocks in \mathcal{A} and picks a random partner that contributes the sites in the other block to the offspring. If $\mathcal{A} = \mathbf{1}$, the reproducing individual contributes all sites. The offspring pieced together in this way replaces a uniformly chosen individual from the population. In this formulation, which is closer in spirit to the deterministic single-crossover model, offspring of type x are created at rate $Nr_{\mathcal{A}}(R_{\mathcal{A}}(Z_t))(x)$ and replace an individual of type y with probability $Z_t(y)/N$. This explains the different normalisation of the original recombinator, whereas the additional factor of $N = \|Z_t\|$ is absorbed in its definition in [9]. The resulting transition rates, however, are again those in (2.7). Here, we stay with the formulation that led to (2.7) in the first place, since it seems more natural for finite populations.*

Let us summarise our model as follows:

Definition 1 (Moran model with single crossovers). *The Moran model with single crossovers is the Markov chain in continuous time $\{Z_t\}_{t \geq 0}$ with state space E of (2.5) and generator matrix Λ with nondiagonal elements*

$$\Lambda(z, z + w) = \sum_{\substack{x, y \in \mathbb{X}: \\ \delta_x - \delta_y = w}} \lambda(z; y, x), \quad w \neq 0,$$

for $z \in E$, $w \in E - z$ (where $E - z := \{v \mid z + v \in E\}$) and $\Lambda(z, z) = - \sum_{\substack{v \in E - z: \\ v \neq 0}} \Lambda(z, z + v)$.

Limits of the forward model. Consider now the family of processes $\{Z_t^{(N)}\}_{t \geq 0}$, $N = 1, 2, \dots$, where we add the upper index to indicate dependence on population size. Also consider the normalised version $\{Z_t^{(N)}/N\}_{t \geq 0}$; $Z_t^{(N)}/N$ is a random probability measure on \mathbb{X} . For $N \rightarrow \infty$ and without any rescaling of the $r_{\mathcal{A}}$ or of time, the sequence $\{Z_t^{(N)}\}_{t \geq 0}$ converges to the solution of the *deterministic single-crossover equation*

$$\dot{\omega}_t = \sum_{\mathcal{A} \in \mathbb{O}_2(S)} r_{\mathcal{A}}(R_{\mathcal{A}}(\omega_t) - \omega_t) \tag{2.8}$$

with initial value ω_0 , ω_0 a probability measure, and we assume that $\lim_{N \rightarrow \infty} Z_0^{(N)}/N = \omega_0$. This is a *dynamical law of large numbers* and due to [30, Thm. 11.2.1]. The precise statement as well as the proof are perfectly analogous to Proposition 1 in [5], which assumes a slightly different sampling scheme for recombination. We therefore leave out the details here. The deterministic single-crossover equation (2.8) was investigated in [9]

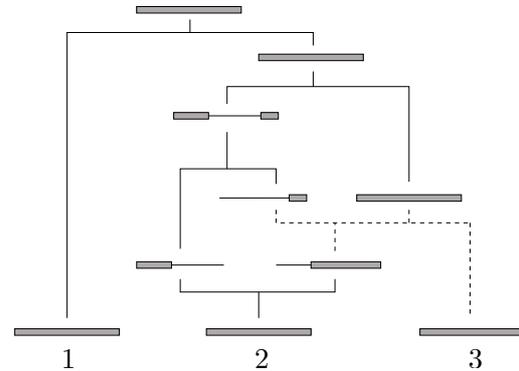
and [8]. For comparison, note that, in view of Remark 2, the probability $r_{\mathcal{A}}$ in (2.8) is multiplied by the unit rate at which each individual reproduces, and this way turns into a recombination *rate*.

The Moran model with recombination also has a well-known diffusion limit, which emerges when $N \rightarrow \infty$ under $Nr_{\mathcal{A}} \rightarrow \varrho_{\mathcal{A}}$, $\varrho_{\mathcal{A}}$ a constant, $\mathcal{A} \in \mathbb{O}_2(S)$, after a speedup of time by a factor of N . In the case of two loci and two alleles, this goes back to [74]; see also [26, Chap. 8.2] for a modern exposition. Two loci with an arbitrary (but finite) number of alleles are treated in [60]. This should readily generalise to the case of a finite number of loci with a finite number of alleles, but we do not spell it out here, since we will not draw on the diffusion limit of the forward process later.

The ancestral recombination process (ARP) and its marginal version. In line with standard population-genetic thinking, we employ a genealogical approach by tracing back the ancestry of (parts of) the genetic material from a population at present that evolved according to the Moran model with single-crossover recombination. The standard genealogical approach for models with recombination is the ancestral recombination graph (ARG) first formulated in [57]. Today, many different notions of ‘ARG’ are in use. We stick to the usual convention here that the ARG assumes the diffusion limit. Hudson’s original version was for two loci, but multilocus generalisations (see [54] and [14]) and continuous sequence versions ($n \rightarrow \infty$, see, e.g., [26, Chap. 3.4]) are immediate. The ARG starts from a sample of individuals from the present population and follows their ancestry backward. When a sequence (or a part of a sequence) experiences a recombination event, it branches into a leading and a trailing segment; when two (parts of) sequences go back to a common ancestor, there is a coalescence event. For overviews see [56, Chap. 5], [26, Chap. 3.4], or [94, Chap. 7.2]. Mutation can be independently superimposed on the ARG, but will not be considered in this thesis. One is then interested in the full information on the sample, namely, the probabilities for all possible type distributions of the sample. The stationary state of the ARG may be characterised by a collection of so-called *sampling recursions*; they may be solved analytically for tiny samples (leading to explicit *sampling formulae*), or numerically for larger ones, see [46], or [26, Chap. 3.6]. But feasibility is limited due to the enormous state space, even for small samples. Alternatively, one resorts to computationally intensive Monte-Carlo or importance-sampling methods to simulate the ARG [54, 95, 61]. Recently, Song and coworkers discovered structural properties of the ARG that allow for an efficient combination of analytical and simulation techniques in the regime of *strong recombination* (see [62]); more precisely, they work in terms of expansions in $1/\varrho$ as $\varrho \rightarrow \infty$, where all $\varrho_{\mathcal{A}}$, with $\mathcal{A} \in \mathbb{O}_2(S)$, are assumed to scale linearly with the common factor ϱ .

In contrast, we will work in the setting of both *finite* n and *finite* N . The corresponding *ancestral recombination process (ARP)*, which is illustrated in Figure 2.2, is a finite-population version of the multilocus ARG. We then simplify matters by only aiming at

Figure 2.2: A realisation of the full ancestral recombination process, starting from $m = 3$ individuals; ancestral material is shaded, non-ancestral material is indicated by thin horizontal lines. The mixed recombination-coalescence event indicated by dashed lines can only appear in the finite population recombination process (ARP). In the diffusion limit, and thus in the ARG, recombination and coalescence act in isolation.



reduced information. Namely, we consider a partition $\mathcal{A} = \{A_1, A_2, \dots, A_m\}$ of S (with $m \leq \min\{n, N\}$). Now sample m individuals from the present population and follow back the ancestry of the sites in A_1 in the first individual, in A_2 in the second individual, \dots , and in A_m in the m 'th individual, without considering any other sites and any other individuals, as in Figure 2.3. That is, each locus is considered in one individual only. The

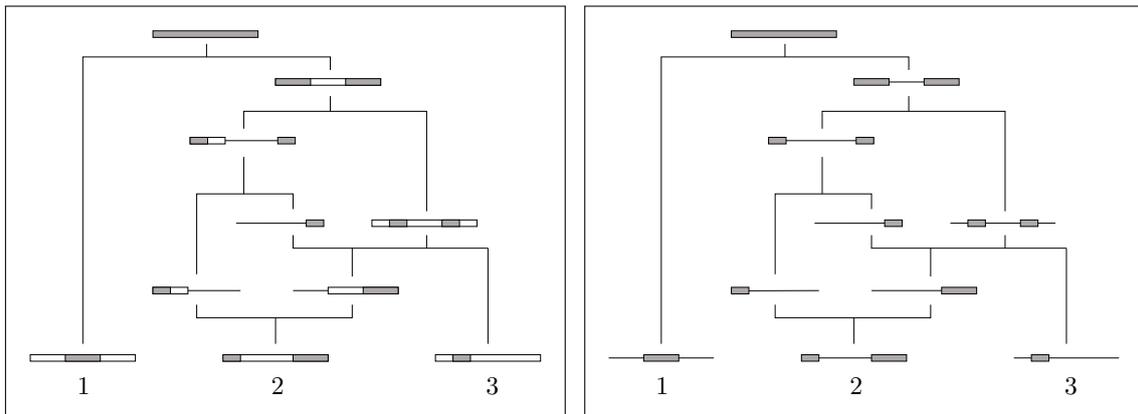


Figure 2.3.: The marginalised version corresponding to the ARP in Figure 2.2, in which we only follow blocks of the partition (shaded), that is, block A_i is sampled in individual number i , $1 \leq i \leq m$. Material that is ancestral to the sampled individuals, but not to the blocks considered, is shown as open rectangles (left). But since this is not traced back, it can be treated in the same way as material non-ancestral to the sampled individuals (right). Consequently, the sample will finally consist of the blocks of the partition only.

result may be viewed as a *marginalised* version of the ancestral recombination process, and, in the diffusion limit, turns into a marginal version of the multilocus ARG starting from a sample of size m . We will see that this information is sufficient to characterise the time evolution of the expected linkage disequilibria of all orders. We will not employ any scaling or limit, in order to allow for arbitrary strengths of recombination. It will turn out

that the approach in [17] actually corresponds to this marginal ancestral recombination process, although this is not apparent from their formulation forward in time.

More precisely, the letters at the loci considered at present, together with their ancestry, can be constructed by a three-step procedure (see Figure 2.4). First, we run a partitioning process $\{\Sigma_t\}_{t \geq 0}$ on $\mathbb{P}(S)$, backward in time, starting at a given initial partition Σ_0 with $|\Sigma_0| = m$. Σ_t describes the partitioning of sites into parental individuals at time t ; sites in the same block (in different blocks) belong to the same (to different) individuals. Clearly, $|\Sigma_t|$ is the number of ancestral individuals at time t . The process $\{\Sigma_t\}_{t \geq 0}$ is independent of the types and will be described in detail in the next section. In the second step, a letter is assigned to each site of S at time t (i.e. in the past) in the following way. For every part of Σ_t , pick an individual from the initial population (without replacement) and copy its letters to the sites in the block considered. For illustration, also assign a colour to each block, thus indicating different parental individuals. In the last step, the letters and colours are propagated downward (i.e. forward in time) according to the realisation of $\{\Sigma_t\}_{t \geq 0}$ laid down in the first step. A similar construction was used in the ancestral process in [7], but restricted to a sample of size 1 (i.e. start with $\Sigma_0 = \mathbf{1}$), and in discrete time in the deterministic limit. Let us now describe the partitioning process in detail.

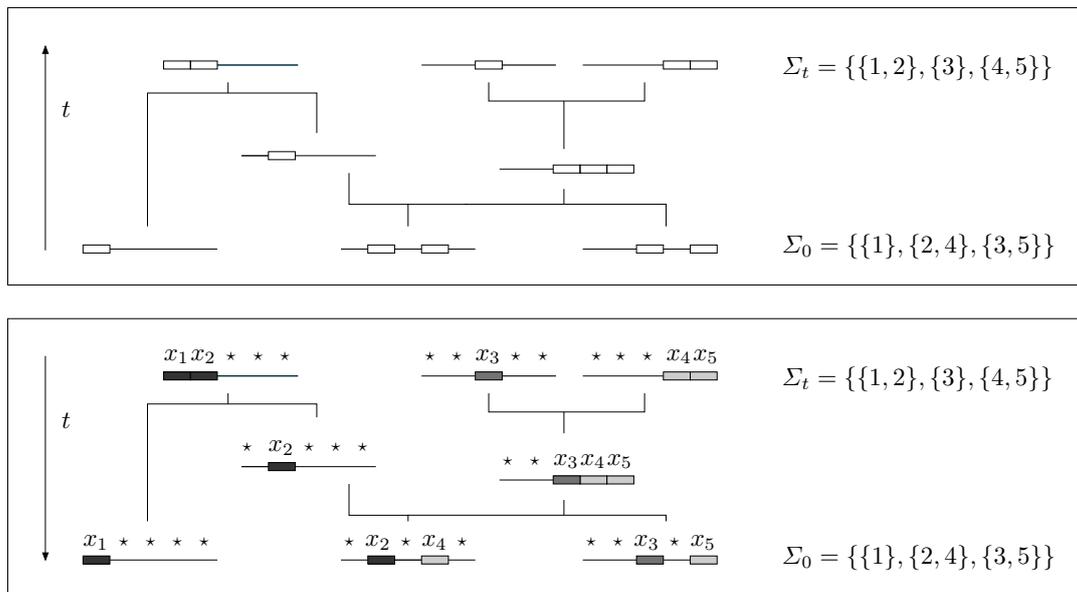


Figure 2.4.: Construction of one possible ancestry of a collection of sites that correspond to the initial partition $\Sigma_0 = \{\{1\}, \{2, 4\}, \{3, 5\}\}$. The upper panel shows the partitioning process (backward in time). In the lower panel, letters and colours are assigned to each block of Σ_t and propagated downward (forward in time).

2.4. The partitioning process

The partitioning process $\{\Sigma_t\}_{t \geq 0}$ is a Markov process on $\mathbb{P}(S)$, which describes how the sites are partitioned into different individuals backward in time. Since there is a one-to-one relationship between the individuals and the blocks of the partition, we may identify individuals with the ancestral material they carry.

The process $\{\Sigma_t\}_{t \geq 0}$ consists of a mixture of splitting (S) and coalescence (C) events. It can be constructed independently of the types. In this section, we describe the process by arguing on the grounds of the underlying Moran model; in Section 2.7, we will formally prove that this is indeed the correct dual process for it.

Since we trace back sites in subsets $U \subseteq S$ (rather than complete sequences), we need the corresponding *marginal recombination probabilities*

$$r_{\mathcal{B}}^U := \sum_{\substack{\mathcal{A} \in \mathbb{O}_{\leq 2}(S) \\ \mathcal{A}|_U = \mathcal{B}}} r_{\mathcal{A}}^S \quad (2.9)$$

for any $\mathcal{B} \in \mathbb{O}_{\leq 2}(U)$, where $r_{\mathcal{A}}^S = r_{\mathcal{A}}$. Note that, for $|U| = 1$, the only recombination parameter is $r_{\mathbf{1}}^U = 1$. If U is ordered in S (i.e. $U = \{x \in S : \min(U) \leq x \leq \max(U)\}$) and $\mathcal{B} \neq \mathbf{1}|_U$, then $r_{\mathcal{B}}^U$ is simply the probability of crossover after the (unique) site that leads to partition \mathcal{B} . If U is not ordered in S , then $r_{\mathcal{B}}^U$ is the sum of the probabilities of all crossovers that lead to partition \mathcal{B} , as illustrated in Figure 2.5.

Assume now that U is an unordered block of Σ_t . This means there is so-called *trapped material*, that is, non-ancestral sites enclosed between ancestral regions. All crossover events within a given trapped segment contribute to the separation of the adjacent ancestral segments – in contrast to crossovers in flanking non-ancestral regions to the left or the right of U , which do not affect the genealogy. Note finally that the upper index in $r_{\mathcal{B}}^U$ can, in principle, be omitted since $U = \cup_{i=1}^{|\mathcal{B}|} B_i$, and we will do so when appropriate.

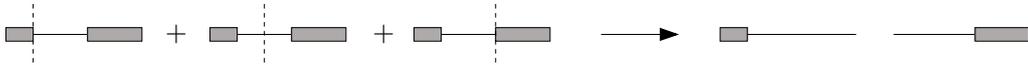


Figure 2.5.: Let $S = \{1, \dots, 5\}$ and $U = \{1, 4, 5\} \subset S$. For the partition $\mathcal{B} = \{\{1\}, \{4, 5\}\}$, there are three recombination events that partition U into \mathcal{B} , thus $r_{\mathcal{B}}^U = r_{\{\{1\}, \{2, 3, 4, 5\}\}} + r_{\{\{1, 2\}, \{3, 4, 5\}\}} + r_{\{\{1, 2, 3\}, \{4, 5\}\}}$.

Now start with the initial partition Σ_0 . Suppose that the current state is $\Sigma_t = \mathcal{A} = \{A_1, \dots, A_m\}$ and denote by Δ the waiting time to the next event. Δ is exponentially distributed with parameter m , since each block corresponds to an individual, and each individual is independently affected at rate 1. When the event happens, choose a block uniformly. If A_j is picked, then $\Sigma_{t+\Delta}$ is obtained as follows (see Figure 2.6 for an example).

In the splitting step, block A_j turns into an intermediate state \mathcal{J} with probability $r_{\mathcal{J}}^{A_j}$, $\mathcal{J} \in \mathbb{O}_{\leq 2}(A_j)$. In detail:

- (S₁) With probability $r_{\mathbf{1}}^{A_j}$, the block A_j remains unchanged. The resulting intermediate state (of this block) is $\mathcal{J} = \mathbf{1}|_{A_j}$.
- (S₂) With probability $r_{\mathcal{J}}^{A_j}$, $\mathcal{J} \in \mathbb{O}_2(A_j)$, block A_j splits into two parts, $\mathcal{J} = \{A_{j_1}, A_{j_2}\}$, which are ordered in A_j , but not necessarily in S . Recall that, via (2.9), $r_{\mathcal{J}}^{A_j}$ takes into account *all* recombination probabilities that lead to \mathcal{J} , including those within trapped material.

Now, each block of \mathcal{J} chooses out of N parents, uniformly and with replacement. Among these, there are $m - 1$ parents that carry one block of $\mathcal{A}_{M \setminus j} = \mathcal{A} \setminus A_j$ each; the remaining $N - (m - 1)$ parents are *empty*, that is, they do not carry ancestral material available for coalescence. Coalescence happens if the choosing block picks a parent that carries ancestral material; otherwise, the choosing block becomes an ancestral block of its own, which is available for coalescence from then onwards. The possible outcomes are certain coarsenings of $\mathcal{A}_{M \setminus j} \cup \mathcal{J}$, namely:

If $\mathcal{J} = \{A_j\}$ (case (S₁)), then either

- (C_{1,1}) With probability $(N - (m - 1))/N$, block A_j does not coalesce with any block of $\mathcal{A}_{M \setminus j}$. As a result, $\Sigma_{t+\Delta} = \Sigma_t = \mathcal{A}$.
- (C_{1,2}) With probability $1/N$, block A_j coalesces with block A_k , $k \in M \setminus j$. This results in $\Sigma_{t+\Delta} = \mathcal{A}_{M \setminus \{j,k\}} \cup A_{\{j,k\}}$.

If $\mathcal{J} = \{A_{j_1}, A_{j_2}\}$ (case (S₂)), we get the following possibilities:

- (C_{2,1}) With probability $(N - (m - 1))(N - m)/N^2$, no block of \mathcal{J} coalesces with a block of $\mathcal{A}_{M \setminus j}$, so $\Sigma_{t+\Delta} = \mathcal{A}_{M \setminus j} \cup \mathcal{J}$.
- (C_{2,2}) With probability $(N - (m - 1))/N^2$, one block of \mathcal{J} coalesces with block A_k , $k \in M \setminus j$, while the other block of \mathcal{J} chooses an empty individual. This ends up in the state $\Sigma_{t+\Delta} = \mathcal{A}_{M \setminus \{j,k\}} \cup \{A_{\{j_1,k\}}, A_{j_2}\}$ or $\Sigma_{t+\Delta} = \mathcal{A}_{M \setminus \{j,k\}} \cup \{A_{\{j_2,k\}}, A_{j_1}\}$. That is, in going from Σ_t to $\Sigma_{t+\Delta}$, either block A_{j_1} or A_{j_2} is moved from A_j to A_k .
- (C_{2,3}) With probability $(N - (m - 1))/N^2$, the blocks A_{j_1} and A_{j_2} coalesce with each other, but choose an empty individual, which gives $\Sigma_{t+\Delta} = \mathcal{A}$.
- (C_{2,4}) With probability $1/N^2$, the block A_{j_1} coalesces with A_k and A_{j_2} coalesces with A_ℓ , $k, \ell \in M \setminus j$. This yields either $\Sigma_{t+\Delta} = \mathcal{A}_{M \setminus \{j,k,\ell\}} \cup \{A_{\{j_1,k\}}, A_{\{j_2,\ell\}}\}$ if $k \neq \ell$, or $\Sigma_{t+\Delta} = \mathcal{A}_{M \setminus \{j,k\}} \cup A_{\{j,k\}}$ if $k = \ell$.

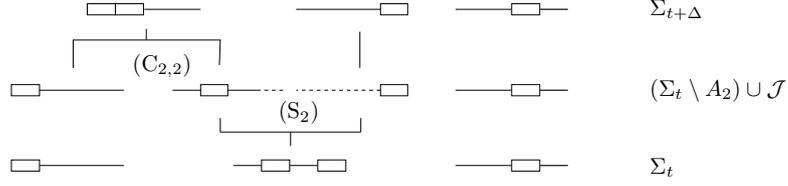


Figure 2.6.: One step of the partitioning process with current state $\Sigma_t = \{A_1, A_2, A_3\} = \{\{1\}, \{2, 4\}, \{3\}\}$. In this example, A_2 is chosen and splits into $\mathcal{J} = \{\{2\}, \{4\}\}$. In the following step $(C_{2,2})$, the leading part coalesces with A_1 , whereas the trailing part remains separate, so that we end up in $\Sigma_{t+\Delta} = \{\{1, 2\}, \{3\}, \{4\}\}$.

Summarising, we see that a transition from \mathcal{A} to \mathcal{B} , via partitioning of block A_j into \mathcal{J} , $j \in M$, $\mathcal{J} \in \mathbb{O}_{\leq 2}(A_j)$, is possible whenever $\mathcal{B} \succcurlyeq \mathcal{A}_{M \setminus j} \cup \mathcal{J}$ and $\mathcal{B}|_{A_{M \setminus j}} = \mathcal{A}_{M \setminus j}$, or, equivalently, whenever

$$\mathcal{B}|_{A_j} \succcurlyeq \mathcal{J} \quad \text{and} \quad \mathcal{B}|_{A_{M \setminus j}} = \mathcal{A}_{M \setminus j}.$$

Each block of \mathcal{J} coalesces into every block currently available with probability $1/N$, and remains separate with probability $(N - k)/N$ if there are currently k blocks available; in the latter case, the block considered becomes number $k + 1$. We can therefore summarise the rate of the said transition as

$$\vartheta_{j, \mathcal{J}; \mathcal{A}, \mathcal{B}} = \begin{cases} r_{\mathcal{J}}^{A_j} \frac{1}{N^{|\mathcal{J}|}} \frac{(N - (m-1))!}{(N - |\mathcal{B}|)!}, & \text{if } \mathcal{B}|_{A_j} \succcurlyeq \mathcal{J}, \mathcal{B}|_{A_{M \setminus j}} = \mathcal{A}_{M \setminus j}, \\ 0, & \text{otherwise.} \end{cases} \quad (2.10)$$

Note that this includes silent events where $\mathcal{B} = \mathcal{A}$. Thus, the partitioning process $\{\Sigma_t\}_{t \geq 0}$ is a continuous-time Markov chain on $\mathbb{P}(S)$ characterised by the generator $\Theta := (\Theta_{AB})_{\mathcal{A}, \mathcal{B} \in \mathbb{P}(S)}$ with nondiagonal elements

$$\begin{aligned} \Theta_{AB} &= \sum_{j \in M} \sum_{\mathcal{J} \in \mathbb{O}_{\leq 2}(A_j)} \vartheta_{j, \mathcal{J}; \mathcal{A}, \mathcal{B}} \\ &= \begin{cases} r_{\mathcal{J}}^{A_j} \frac{1}{N^2} \frac{(N - (m-1))!}{(N - |\mathcal{B}|)!}, & \text{if } \mathcal{B}|_{A_j} = \mathcal{J}, \mathcal{B}|_{A_{M \setminus j}} = \mathcal{A}_{M \setminus j}, \\ & \text{for some } j \in M, \mathcal{J} \in \mathbb{O}_2(A_j), \\ \frac{2}{N^2} + \frac{N-1}{N^2} (r_{\mathbf{1}}^{A_j} + r_{\mathbf{1}}^{A_k}), & \text{if } \mathcal{B} = \mathcal{A}_{M \setminus \{j, k\}} \cup A_{\{j, k\}} \\ & \text{for some } j \neq k \in M, \\ 0, & \text{for all other } \mathcal{B} \neq \mathcal{A}. \end{cases} \end{aligned} \quad (2.11)$$

Note that, for $\mathcal{J} \in \mathbb{O}_2(A_j)$ we have distinguished between $\mathcal{B}|_{A_j} = \mathcal{J}$ and $\mathcal{B}|_{A_j} = \mathbf{1}|_{A_j} \succcurlyeq \mathcal{J}$. The latter corresponds to $k = \ell$ in $(C_{2,4})$ and leads to the same transition as a *pure coalescence event* in $(C_{1,2})$. More precisely, the total coalescence rate of j and k is

$$\frac{1}{N} (r_{\mathbf{1}}^{A_j} + r_{\mathbf{1}}^{A_k}) + \frac{1}{N^2} \left(\sum_{\mathcal{J} \in \mathbb{O}_2(A_j)} r_{\mathcal{J}}^{A_j} + \sum_{\mathcal{K} \in \mathbb{O}_2(A_k)} r_{\mathcal{K}}^{A_k} \right) = \frac{2}{N^2} + \frac{N-1}{N^2} (r_{\mathbf{1}}^{A_j} + r_{\mathbf{1}}^{A_k}) \quad (2.12)$$

as stated, since

$$\sum_{\mathcal{J} \in \mathbb{O}_2(U)} r_{\mathcal{J}}^U = 1 - r_{\mathbf{1}}^U, \quad (2.13)$$

$U \subseteq S$. Moreover, transitions to partitions \mathcal{B} with $|\mathcal{B}| > N$ are impossible, as it must be.

Corollary 1. *The diagonal elements of the generator defined in (2.11) are given by*

$$-\Theta_{\mathcal{A}\mathcal{A}} = \frac{m(m-1)}{N} + \frac{(N-1)(N-(m-1))}{N^2} r_{tot}(\mathcal{A}), \quad (2.14a)$$

for $m = |\mathcal{A}|$ and

$$r_{tot}(\mathcal{A}) = \sum_{j \in M} \sum_{\mathcal{J} \in \mathbb{O}_2(A_j)} r_{\mathcal{J}}^{A_j} = w_{\mathcal{A}}^T r_{\mathcal{B}}^S, \quad \text{where } w_{\mathcal{A}} = \left(\sum_{j \in M} \delta_{\mathcal{B}|_{A_j} \in \mathbb{O}_2(A_j)} \right)_{\mathcal{B} \in \mathbb{O}_2(S)} \quad (2.14b)$$

Proof. Using (2.9) with $\mathcal{B} = \mathcal{A}$ and the fact that it is the diagonal element of a Markov generator, then separating the case $\mathcal{J} = \mathbf{1}$ yields

$$\begin{aligned} \Theta_{\mathcal{A}\mathcal{A}} &= - \sum_{\mathcal{B} \in \mathbb{P}(S) \setminus \mathcal{A}} \Theta_{\mathcal{A}\mathcal{B}} = \sum_{j \in M} \left[\sum_{\mathcal{J} \in \mathbb{O}_{\leq 2}(A_j)} \vartheta_{j, \mathcal{J}; \mathcal{A}, \mathcal{A}} - 1 \right] \\ &= \sum_{j \in M} \left[\left(\frac{1}{N} r_{\mathbf{1}}^{A_j} + \frac{1}{N^2} \sum_{\mathcal{J} \in \mathbb{O}_2(A_j)} r_{\mathcal{J}}^{A_j} \right) (N - (m-1)) - 1 \right], \end{aligned}$$

and with (2.13), expanding the inner product, and rearranging the sum, we obtain

$$= - \left[\frac{m(m-1)}{N} + \frac{(N-1)(N-(m-1))}{N^2} \sum_{j \in M} \sum_{\mathcal{J} \in \mathbb{O}_2(A_j)} r_{\mathcal{J}}^{A_j} \right]. \quad \square$$

Remark 3. *The second representation of $r_{tot}(\mathcal{A})$ in (2.14b) is a simple linear combination of the basic recombination probabilities $r_{\mathcal{B}}^S$, $\mathcal{B} \in \mathbb{O}_2(S)$, where the elements of the (column) vector $w_{\mathcal{A}}$ specify how many blocks $A \in \mathcal{A}$ are affected by each simple split denoted by $\mathcal{B} \in \mathbb{O}_2(S)$. This representation is mainly used in Algorithms 2 and 3 in Section 3.4. Note that $r_{tot}(\mathcal{A})$ is not a probability anymore and vanishes for $\mathcal{A} = \mathbf{0}$.*

Now we have a closer look at the number of possible non-zero transitions. For a given $\mathcal{A} \in \mathbb{P}(S)$, the following types are possible with the corresponding numbers of occurrence:

$$\mathcal{A} \rightarrow \mathcal{A} : \quad 1, \quad (2.15a)$$

$$\mathcal{A} \rightarrow \mathcal{A}_{M \setminus j} \cup A_{\{j,k\}}, j \neq k : \quad \binom{m}{2}, \quad (2.15b)$$

$$\mathcal{A} \rightarrow \mathcal{A}_{M \setminus j} \cup \mathcal{J} : \quad \sum_{j=1}^m (|A_j| - 1), \quad (2.15c)$$

$$\mathcal{A} \rightarrow \mathcal{A}_{M \setminus \{j,k,l\}} \cup \{A_{\{j_1,k\}} \cup A_{\{j_2,l\}}\}, j \neq k \neq l : \quad \sum_{j=1}^m (|A_j| - 1)(m-1)(m-2), \quad (2.15d)$$

# sites (n)	# states (b_n)	# (ordered) state pairs	# non-zero transitions	ratio
2	2	4	4	1.000
3	5	25	22	0.880
4	15	225	133	0.591
5	52	2704	831	0.307
6	203	41209	5331	0.129
7	877	769129	35239	0.045
8	4140	17129600	240878	0.014
9	21147	447195609	1706252	0.004

Table 2.1.: Number of non-zero transitions and its ratio to the number of ordered state pairs for $n = 2, 3, \dots, 9$.

$$\mathcal{A} \rightarrow \mathcal{A}_{M \setminus \{j,k\}} \cup \{A_{\{j_1,k\}} \cup A_{j_2}\}, j \neq k : \quad \sum_{j=1}^m (|A_j| - 1)(m - 1), \quad (2.15e)$$

$$\mathcal{A} \rightarrow \mathcal{A}_{M \setminus \{j,k\}} \cup \{A_{j_1} \cup A_{\{j_2,k\}}\}, j \neq k : \quad \sum_{j=1}^m (|A_j| - 1)(m - 1). \quad (2.15f)$$

Note, that $\sum_{j=1}^m (|A_j| - 1) = n - m$ is independent on the size or structure of the particular blocks. So, a partition \mathcal{A} allows for

$$1 + \binom{m}{2} + (n - m) \left(1 + (m - 1)(m - 2) + 2(m - 1) \right) = 1 + (n - m) + \binom{m}{2} (2(n - m) + 1) \quad (2.16)$$

non-zero transitions in total. Therefore, the overall amount of those transitions described by Θ is

$$\sum_{m=1}^n \left\{ \begin{matrix} n \\ m \end{matrix} \right\} \left(\binom{m}{2} (2(n - m) + 1) + (n - m) \right) + b_n, \quad (2.17)$$

where $b_n := |\mathbb{P}(S)|$ denotes the n -th *Bell number*, i.e. the number of possible partitions of the set S with $n = |S|$ (cf. [13, Sec. 1.11] and [1, Ex. II.4.E]), and thus the number of rows (columns) of Θ . Furthermore, $\left\{ \begin{matrix} n \\ m \end{matrix} \right\}$ denotes the (*unsigned*) *Stirling number of second kind*, which is the number of ways to partition S into m non-empty blocks. Hence, the latter is related to the n -th Bell number through $b_n = \sum_{m=0}^n \left\{ \begin{matrix} n \\ m \end{matrix} \right\}$. In Table 2.1 we state the number of non-zero transitions and its ratio to the number of ordered state pairs for $n = 2, 3, \dots, 9$.

Connection with previous work by Bobrowski et al. In fact, the generator Θ coincides with the generator worked out by Bobrowski et al. in [16] and [17] with a very different approach, forward in time. For $n \leq 3$, they state the generator matrix explicitly, and identity (2.11) is easily checked by elementwise comparison. For $n > 3$, they provide an algorithm, which runs through all individuals and all sites and builds up the matrix Θ incrementally, in the following manner. For every given individual, leading and

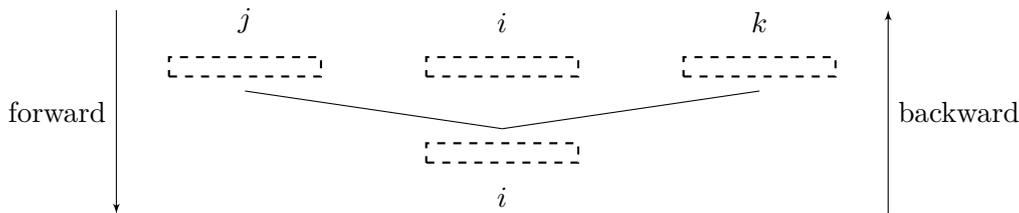


Figure 2.7.: Correspondance of the transitions of the partitioning process (backward in time) and the ones in [17] (forward in time). A replacement of individual i forward in time with leading part (up to some site $s \in S \setminus n$) from j and trailing part (beginning with site $s + 1$) from k means a split of individual i between site s and $s + 1$ followed by a coalescence of its leading part i_1 with individual j and its trailing part i_2 with individual k .

trailing segments (for the split between site s and $s + 1$, for all $s \in S \setminus n$) are taken into account, irrespective of whether or not the segments contain ancestral material. This way, their algorithm does not distinguish between transitions induced by recombination events within ancestral (or trapped) material and recombination events that are invisible in the genealogical perspective, that is, those that are effectively pure coalescence events. Instead, a case distinction is made that is based on whether (or not) one (or both) segment(s) coalesce with individuals that do (or do not) carry ancestral material. We investigate this approach and show the connection to ours by translating corresponding model parameters, expanding their 5 cases (without the self transitions) into 11 subcases, and a following rearrangement according to the emerging partitions of the complete ancestral material. This leads precisely to our cases $(C_{2,1})$ to $(C_{2,4})$ (here, both emerging segments contain ancestral material) as well as $(C_{1,1})$ and $(C_{1,2})$ (here one segment is empty).

First, their population is of size $2N$ and they model the individuals' lifespan as exponentially distributed with parameter $\lambda/2$. To translate the model in our scaling, we have to multiply every transition probability by $2N$ and set $\lambda = 1$. Then, we can also set N as the model population size in their setting. Furthermore, they decompose

$$\Theta = \sum_{\mathcal{A} \in \mathbb{O}_{\leq 2}(S)} r_{\mathcal{A}} \Theta_{\mathcal{A}}, \quad (2.18)$$

so they separate the transition rates into parts with respect to the position between two consecutive sites, and construct each (hence accruing) $\Theta_{\mathcal{A}}$ *separately* in the following manner: For a transition from fixed \tilde{i} (which corresponds to \mathcal{A} in our notation, so we can also identify $|\mathcal{A}| = m$ with their μ), they go through all possible *recombination events*, identify the correct ι (a possible \mathcal{B}), and add the corresponding probability of occurrence to the generator entry. By a recombination event forward in time they mean the event that an individual i deceases and is replaced partly by j and k . This corresponds to a splitting and coalescence transition backward in time where A_i splits between site $s \in S \setminus n$ and $s + 1$. Then, the leading and trailing parts coalesce with j and k , cf. Figure 2.7.

case	transition in [17]	probability	i_1 & i_2 ancestral	only i_2 ancestral
2.1	$i = j = k$		$\mathcal{A}_{M \setminus i} \cup A_{\{i_1, i_2\}} = \mathcal{A}$	\mathcal{A}
2.2	$i = k, j \neq k$		$\mathcal{A}_{M \setminus \{i, j\}} \cup A_{i_2} \cup A_{\{j, i_1\}}$	\mathcal{A}
2.3	$i = k, i \neq j$	N^{-2}	$\mathcal{A}_{M \setminus \{i, k\}} \cup A_{i_1} \cup A_{\{k, i_2\}}$	$\mathcal{A}_{M \setminus \{i, k\}} \cup A_{\{k, i\}}$
2.4	$i \neq j, i \neq k, j \neq k$		$\mathcal{A}_{M \setminus \{i, j, k\}} \cup A_{\{j, i_1\}} \cup A_{\{k, i_2\}}$	$\mathcal{A}_{M \setminus \{i, k\}} \cup A_{\{k, i\}}$
2.5	$i \neq j, i \neq k, j = k$		$\mathcal{A}_{M \setminus \{i, j\}} \cup A_{\{j, i\}}$	$\mathcal{A}_{M \setminus \{i, j\}} \cup A_{\{j, i\}}$
3.1	$i = j, k > \mu$	$\frac{N-m}{N^2}$	$\mathcal{A}_{M \setminus i} \cup A_{i_1} \cup A_{i_2}$	\mathcal{A}
3.2	$i \neq j, k > \mu$		$\mathcal{A}_{M \setminus \{i, j\}} \cup A_{\{i_1, j\}} \cup A_{i_2}$	\mathcal{A}
4.1	$i = k, j > \mu$	$\frac{N-m}{N^2}$	$\mathcal{A}_{M \setminus i} \cup A_{i_2} \cup A_{i_1}$	\mathcal{A}
4.2	$i \neq k, j > \mu$		$\mathcal{A}_{M \setminus \{i, k\}} \cup A_{\{k, i_2\}} \cup A_{i_1}$	$\mathcal{A}_{M \setminus \{i, k\}} \cup A_{\{k, i\}}$
5	$j = k > \mu$	$\frac{N-m}{N^2}$	$\mathcal{A}_{M \setminus i} \cup A_{\{i_1, i_2\}} = \mathcal{A}$	\mathcal{A}
6	$\mu < j \neq k > \mu$	$\frac{(N-m)(N-(m+1))}{N^2}$	$\mathcal{A}_{M \setminus i} \cup A_{i_1} \cup A_{i_2}$	\mathcal{A}

Table 2.2.: Transitions of the partitioning process connected with the approach in [17]. The leading part i_1 always corresponds to j and the trailing part i_2 corresponds to k . In column 1 and 2 we use the notation of [17] and in column 3 to 5 we use ours. In the cases 3 to 6 the condition $j > \mu$ ($k > \mu$) means, in our interpretation, that individual j (k) does not carry any ancestral material.

Note again that it is not distinguished in the first step whether or not the parts contain ancestral material. This is done in [17] by a sixfold case distinction (where the first one only deals with the self transitions, so we omit this here), whose cases 2 to 6 are split up even further in Table 2.2. As we can see, if both parts i_1 and i_2 contain ancestral material, then our transitions based on (S_2) are exactly composed of in the following manner:

$$(C_{2,1}) \text{ of } 6, 3.1, 4.1, \quad (C_{2,2}) \text{ of } 3.2, 2.2(2.3), \quad (C_{2,3}) \text{ of } 5, 2.1, \quad \text{and} \quad (C_{2,4}) \text{ of } 2.4, 2.5;$$

and if only one part contains ancestral material, then the compositions are:

$$(C_{1,1}) \text{ of } 2.1, 2.2, 3.1, 3.2, 4.1, 5, 6, \quad \text{and} \quad (C_{1,2}) \text{ of } 2.3, 2.4, 2.5, 4.2.$$

Summarising, the approach of Bobrowski et al. disguises or mixes the various partitions of ancestral material that may arise due to a transition, and therefore does not lead to a closed expression for Θ . In contrast, our approach yields the matrix elements explicitly for arbitrary n , and gives them a natural and plausible meaning in terms of the partitioning process in backward time.

Limits of the partitioning process. We now examine how the partitioning process behaves in the two limiting cases mentioned in Section 2.3, namely, the deterministic limit and the diffusion limit. Recall that, in the deterministic limit, we let $N \rightarrow \infty$ without rescaling the recombination probabilities or time. Consider, therefore, the family of processes $\{\Sigma_t^{(N)}\}_{t \geq 0}$, $N = 1, 2, \dots$, generated by $\Theta^{(N)}$, where we again make the

dependence on population size explicit through the upper index. In the limit, only the pure splitting events ($C_{2,1}$) survive, more precisely:

Proposition 1 (Deterministic limit). *In the deterministic limit, the sequence of partitioning processes $\{\Sigma_t^{(N)}\}_{t \geq 0}$ with initial states $\Sigma_0^{(N)} \equiv \sigma$ converges in distribution to the process $\{\Sigma_t'\}_{t \geq 0}$ with initial state $\Sigma_0' = \sigma$ and generator Θ' defined by its nondiagonal elements*

$$\Theta'_{AB} = \begin{cases} r_{\mathcal{J}}^{A_j}, & \text{if } \mathcal{B} = \mathcal{A}_{M \setminus j} \cup \mathcal{J} \text{ for some } j \in M \text{ and } \mathcal{J} \in \mathbb{O}_2(A_j), \\ 0, & \text{for all other } \mathcal{B} \neq \mathcal{A}. \end{cases}$$

Hence, $\{\Sigma_t'\}_{t \geq 0}$ is a process of progressive refinements, that is, $\Sigma_\tau' \preceq \Sigma_t'$ for all $\tau > t$. In particular, if $\Sigma_0' \in \mathbb{O}(S)$, then $\Sigma_t' \in \mathbb{O}(S)$ for all times.

Proof. Inspecting the N -dependence of the elements of $\Theta = \Theta^{(N)}$ in (2.11) gives the following order of magnitude for the nondiagonal elements:

$$\Theta_{AB}^{(N)} = \begin{cases} \frac{1}{N^{m+1-|\mathcal{B}|}} r_{\mathcal{J}}^{A_j} (1 + \mathcal{O}(\frac{1}{N})), & \text{if } \mathcal{B}|_{A_j} = \mathcal{J}, \mathcal{B}|_{A_{M \setminus j}} = \mathcal{A}_{M \setminus j} \\ & \text{for some } j \in M, \mathcal{J} \in \mathbb{O}_2(A_j), \\ \frac{1}{N} (r_{\mathbf{1}}^{A_j} + r_{\mathbf{1}}^{A_k}) + \mathcal{O}(\frac{1}{N^2}), & \text{if } \mathcal{B} = \mathcal{A}_{M \setminus \{j,k\}} \cup A_{\{j,k\}} \\ & \text{for some } j \neq k \in M, \\ 0, & \text{for all other } \mathcal{B} \neq \mathcal{A}. \end{cases} \quad (2.19)$$

Obviously, $\Theta^{(N)} = \Theta' + \mathcal{O}(1/N)$, which proves convergence of the sequence of generators of $\{\Sigma_t^{(N)}\}_{t \geq 0}$ to that of $\{\Sigma_t'\}_{t \geq 0}$. This entails convergence of the corresponding sequence of semigroups. With the help of Theorems 4.2.11 and 4.9.10 in [30], this guarantees convergence of $\{\Sigma_t^{(N)}\}_{t \geq 0}$ to $\{\Sigma_t'\}_{t \geq 0}$ in distribution.

The remainder of the statement is obvious since, under Θ' , the only transitions are those that involve the refinement of a single block, say A_j , into two blocks ordered in A_j . If Σ_0' is ordered in S , then all its blocks are ordered in S , and all blocks of Σ_t' will be ordered in S for all times. \square

Remark 4. *Obviously, in the limit, ancestral material that has been separate will never come together again in one individual, such that there are no coalescence events. When starting with $\Sigma_0' = \{S\}$, the genealogy may be represented by a binary tree, which successively branches into smaller segments; for other initial conditions, one gets a corresponding collection (i.e. a forest) of binary trees. We call these trees ancestral recombination trees or ARTs; a discrete-time analogue was studied in [7].*

We now turn to the diffusion limit and use the factor N rather than (the more common) $2N$ since our N is the *haploid* population size. Here, one considers a sequence of processes in which time is sped up by a factor of N and the recombination probabilities $r_{\mathcal{A}}$ are rescaled such that $\lim_{N \rightarrow \infty} N r_{\mathcal{A}} \rightarrow \varrho_{\mathcal{A}}$, $\varrho_{\mathcal{A}}$ a constant, for $\mathcal{A} \in \mathbb{O}_2(S)$; consequently, $r_{\mathbf{1}} \rightarrow 1$ as

$N \rightarrow \infty$. Note that the $\varrho_{\mathcal{A}}$ are *rates* rather than probabilities. The corresponding ARG is the obvious generalisation of Hudson's original ARG to n loci, which we formulate here in our framework for the sake of completeness, as follows. Every ordered pair of lines coalesces at rate 1; every line splits into two at rate $\varrho_{\mathcal{A}}$ for every $\mathcal{A} \in \mathbb{O}_2(S)$, and the ancestral material is distributed between the new lines according to \mathcal{A} .

In this formulation, however, certain silent events are included, namely those events that happen in non-ancestral material flanking the ancestral parts. These events do not affect the partitioning of ancestral material and may be removed by working with the marginalised recombination rates instead. That is, if a sequence currently carries a set U of ancestral sites, then the relevant recombination rates (in the diffusion limit) are $\varrho_{\mathcal{B}}^U$, with $\mathcal{B} \in \mathbb{O}_2(U)$, which are defined as in (2.9) but with r replaced by ϱ . Analogous modifications where the recombination rates depend on the (continuous) region spanned by ancestral material have been investigated in [98] as well as [70].

If we now restrict attention to the ancestry of n loci partitioned between m individuals, we obtain the *marginal version* of the ARG, which may be formulated as follows.

Definition 2 (Marginalised n -locus ARG). *Start with the set of n sites distributed across $m \leq n$ individuals (or lines) according to a partition Σ_0'' with m parts. Throughout the process, every line is identified with the ancestral material it carries. If it currently carries ancestral sites $U \subseteq S$, it splits into $\mathcal{J} \in \mathbb{O}_2(U)$ at rate $\varrho_{\mathcal{J}}^U$. Every ordered pair of lines coalesces at rate 1, and so do the ancestral sites they carry. That is, the marginalised ARG is the partitioning process $\{\Sigma_t''\}_{t \geq 0}$ defined by the generator Θ'' with nondiagonal elements*

$$\Theta''_{\mathcal{A}\mathcal{B}} = \begin{cases} \varrho_{\mathcal{J}}^{A_j}, & \text{if } \mathcal{B} = \mathcal{A}_{M \setminus j} \cup \mathcal{J} \text{ for some } j \in M, \mathcal{J} \in \mathbb{O}_2(A_j), \\ 2, & \text{if } \mathcal{B} \succ \mathcal{A} \text{ and } |\mathcal{B}| = |\mathcal{A}| - 1, \\ 0, & \text{for all other } \mathcal{B} \neq \mathcal{A}. \end{cases}$$

Proposition 2 (Diffusion limit of the partitioning process). *In the diffusion limit, the sequence of partitioning processes $\{\Sigma_{Nt}^{(N)}\}_{t \geq 0}$ with initial states $\Sigma_0^{(N)} \equiv \sigma$ converges in distribution to the process $\{\Sigma_t''\}_{t \geq 0}$ with initial state $\Sigma_0'' = \sigma$ and generator Θ'' .*

Proof. Due to the rescaling of time, the generator of $\{\Sigma_{Nt}^{(N)}\}_{t \geq 0}$ has nondiagonal elements $N\Theta_{\mathcal{A}\mathcal{B}}^{(N)}$. Referring back to (2.19), they converge to $\lim_{N \rightarrow \infty} N\Theta_{\mathcal{A}\mathcal{B}}^{(N)} = \Theta''_{\mathcal{A}\mathcal{B}}$, since we have $r_{\mathbf{1}}^U \rightarrow 1$ and $Nr_{\mathcal{J}}^U \rightarrow \varrho_{\mathcal{J}}^U$ for $\mathcal{J} \in \mathbb{O}_2(U)$. With the same argument as in the proof of Proposition 1, one obtains convergence in distribution as claimed. \square

Remark 5. *As was to be expected, only pure splitting events and pure coalescence events survive in the diffusion limit. The 'mixed transitions', which involve both splitting and coalescence (i.e. the dashed lines in Figure 2.2) vanish under the rescaling; see also [56, Fig. 5.11]. Let us note that several other variants of the recombination process lead to the same diffusion limit. For example, this is true of the simpler (but biologically less realistic)*

versions of the continuous-time Moran model with recombination where recombination is a parallel process that happens independently of reproduction (rather than coupled to reproduction as assumed here), see [5]. Even the discrete-time Wright-Fisher model with recombination lies in the domain of attraction of the diffusion limit.

2.5. Recombinators and sampling functions

In this section, we will have a closer look at two operators associated with recombination and how they are related to each other. We start by generalising our recombinators and then we introduce closely related sampling functions. In Section 2.6 this analysis will be continued and the group of new operators will be enhanced by multilocus correlation functions.

Recombinators. We have already met $R_{\mathcal{A}}$ for $\mathcal{A} \in \mathbb{O}_{\leq 2}(S)$; we now need the generalisation to arbitrary $\mathcal{A} \in \mathbb{P}(S)$. For $\omega \in \mathcal{M}_+(\mathbb{X}) \setminus 0$, we first define the *non-normalised recombinator* via

$$\bar{R}_{\mathcal{A}}(\omega) = (\pi_{A_1} \cdot \omega) \otimes \cdots \otimes (\pi_{A_m} \cdot \omega), \quad (2.20)$$

where it is implied that the product measure refers to the ordering of the sites as specified by the set S . In words, $\bar{R}_{\mathcal{A}}$ turns ω into the product of its marginals with respect to the blocks in \mathcal{A} . We will throughout denote non-normalised mappings by an overbar. Clearly, $\bar{R}_{\emptyset}(\omega) = \|\omega\|$, $\bar{R}_{\mathbf{1}}(\omega) = \omega$ and $\|\bar{R}_{\mathcal{A}}(\omega)\| = \|\omega\|^{|\mathcal{A}|}$. The corresponding normalised version is

$$R_{\mathcal{A}}(\omega) := \frac{\bar{R}_{\mathcal{A}}(\omega)}{\|\bar{R}_{\mathcal{A}}(\omega)\|}, \quad (2.21)$$

which is well-defined since $\omega \neq 0$. Obviously, $R_{\mathcal{A}}(\omega) = \bar{R}_{\mathcal{A}}(\omega/\|\omega\|)$ and $R_{\mathcal{A}}(\omega)$ is a probability measure on \mathbb{X} , which coincides with (2.6) for $\mathcal{A} \in \mathbb{O}_{\leq 2}(S)$.

Let us now give a probabilistic interpretation for the case that a recombinator $R_{\mathcal{A}}$ acts on a certain population described by a counting measure $z \in E$. For the moment, attach labels from the collection $\mathbb{L} := \{1, 2, \dots, N\}$ to the N individuals in the population in a one-to-one manner, and let these individuals have (random) types $X_t^1, X_t^2, \dots, X_t^N \in \mathbb{X}$ at time t . The type distribution then is $Z_t = \sum_{k=1}^N \delta_{X_t^k}$. For $U \subseteq S$ and $k \in \mathbb{L}$, let $X_{t,U}^k := \pi_U(X_t^k)$, and consider the following procedure. Let a partition $\mathcal{A} = \{A_1, \dots, A_m\}$ of S together with a collection of labels $\ell = (\ell_1, \dots, \ell_m) \in \mathbb{L}^m$ associated with the blocks be given, i.e. $(\mathcal{A}, \ell) := \{(A_1, \ell_1), \dots, (A_m, \ell_m)\}$. Then, piece together a sequence by taking the sites in A_1 from individual ℓ_1 , the sites in A_2 from individual ℓ_2 , \dots , and the sites in A_m from individual ℓ_m . The resulting sequence is $X_{t,\mathcal{A}}^\ell := (X_{t,A_1}^{\ell_1}, \dots, X_{t,A_m}^{\ell_m})$. Now, let $\mathcal{L} \in \mathbb{L}^m$ be an m -fold random drawing *with replacement* from \mathbb{L} , i.e. $\mathcal{L}_1, \dots, \mathcal{L}_m$ are independent and identically distributed uniform random variables with support \mathbb{L} . We

are now interested in the (random) sequence

$$X_{t,\mathcal{A}} := X_{t,\mathcal{A}}^{\mathcal{L}}$$

and the corresponding counting measure

$$|\{X_{t,\mathcal{A}} = x\}| = |\{\ell \in \mathbb{L}^m \mid X_{t,\mathcal{A}}^\ell = x\}|.$$

This counts how often one obtains sequence x when performing the above procedure on a population Z_t and all combinations of individuals are included. Since we draw the labels \mathcal{L}_j with replacement (and therefore independently), we can decompose the counting measure

$$|\{X_{t,\mathcal{A}} = x\}| = \prod_{j \in M} |\{\ell_j \in \mathbb{L} \mid X_{t,A_j}^{\ell_j} = x_{A_j}\}| = \prod_{j \in M} (\pi_{A_j} \cdot Z_t)(x_{A_j}) = (\bar{R}_{\mathcal{A}}(Z_t))(x). \quad (2.22)$$

Clearly, $R_{\mathcal{A}}(Z_t)$, the corresponding normalised version, is the type distribution that results when a sequence is created by taking the letters for the blocks in \mathcal{A} from individuals drawn uniformly and with replacement from the population Z_t . So

$$(R_{\mathcal{A}}(z))(x) = \mathbf{P}[X_{t,\mathcal{A}} = x \mid Z_t = z],$$

where \mathbf{P} denotes probability. Note that the left-hand side depends on time only through the value z of Z_t .

Sampling function. For $\mathcal{A} \in \mathbb{P}(S)$ and $\omega \in \mathcal{M}_+(\mathbb{X}) \setminus 0$, we now define our *sampling function*

$$\bar{H}_{\mathcal{A}}(\omega) := \sum_{\mathcal{B} \succ \mathcal{A}} \mu(\mathcal{A}, \mathcal{B}) \bar{R}_{\mathcal{B}}(\omega), \quad (2.23)$$

where μ is the Möbius function in (2.2). $\bar{H}_{\mathcal{A}}(\omega)$ is not a positive measure in general; but it will turn out as positive for the important case where $\omega \in E$ with $\|\omega\| \geq |\mathcal{A}|$, see Lemma 1. We will therefore postpone the normalisation step. In any case, Möbius inversion (compare (2.3) and (2.4)) immediately yields the inverse of (2.23):

Fact 1. For every $\mathcal{A} \in \mathbb{P}(S)$,

$$\bar{R}_{\mathcal{A}}(\omega) = \sum_{\mathcal{B} \succ \mathcal{A}} \bar{H}_{\mathcal{B}}(\omega). \quad \square$$

We can now give $\bar{H}_{\mathcal{A}}$ a meaning by reconsidering the procedure that led to (2.22) but, this time, individuals are not replaced. Therefore, let $|\mathcal{A}| = m \leq N$ and $\tilde{\mathcal{L}} \in \mathbb{L}^m$ be an m -fold random drawing *without replacement* from \mathbb{L} , i.e. $\tilde{\mathcal{L}}_1, \dots, \tilde{\mathcal{L}}_m$ are now *dependent*. Then we look at

$$\tilde{X}_{t,\mathcal{A}} := X_{t,\mathcal{A}}^{\tilde{\mathcal{L}}} \quad (2.24)$$

with the corresponding counting measure $|\{\tilde{X}_{t,\mathcal{A}} = x\}|$. Since individuals are not replaced,

an expression for $|\{\tilde{X}_{t,\mathcal{A}} = x\}|$ analogous to (2.22) is not immediate. Instead, we resort to an inclusion-exclusion argument and prove

Proposition 3. *For $\mathcal{A} \in \mathbb{P}(S)$ with $|\mathcal{A}| = m \leq N$ and $Z_t \in E$, we have*

$$|\{\tilde{X}_{t,\mathcal{A}} = x\}| = |\{\ell \in \mathbb{L}^m \mid X_{t,\mathcal{A}}^\ell = x \text{ and } \ell_i \neq \ell_j \forall i \neq j\}| = (\bar{H}_{\mathcal{A}}(Z_t))(x).$$

Proof. Fix a given partition $\mathcal{A} \in \mathbb{P}(S)$ with $|\mathcal{A}| = m \leq N$. For every $\ell \in \{1, 2, \dots, N\}^m$, the pair (\mathcal{A}, ℓ) uniquely defines a pair $(\mathcal{B}, \tilde{\ell})$, where $\tilde{\ell} \in \{\ell \in \{1, 2, \dots, N\}^{|\mathcal{B}|} \mid \ell_i \neq \ell_j \forall i \neq j\}$ and $\mathcal{B} \succcurlyeq \mathcal{A}$, as follows. Join all blocks of \mathcal{A} that have the same label, and attach that label to the new block. The result is $(\mathcal{B}, \tilde{\ell})$. The other way round, every $(\mathcal{B}, \tilde{\ell})$ with $\mathcal{B} \succcurlyeq \mathcal{A}$ and $\tilde{\ell} \in \{\ell \in \{1, 2, \dots, N\}^{|\mathcal{B}|} \mid \ell_i \neq \ell_j \forall i \neq j\}$ uniquely defines the labelling ℓ of the blocks of \mathcal{A} (keep in mind that \mathcal{A} is fixed): block $A_k \in \mathcal{A}$ receives the label of that block $B_j \in \mathcal{B}$ in which it is contained. We can therefore identify the set $\{(\mathcal{A}, \ell) \mid \ell \in \{1, 2, \dots, N\}^m\}$ with the set $\bigcup_{\mathcal{B} \succcurlyeq \mathcal{A}} \{(\mathcal{B}, \tilde{\ell}) \mid \tilde{\ell} \in \{\ell \in \{1, 2, \dots, N\}^{|\mathcal{B}|} \mid \ell_i \neq \ell_j \forall i \neq j\}\}$ and *decompose* the event $\{X_{t,\mathcal{A}} = x\} = \dot{\bigcup}_{\mathcal{B} \succcurlyeq \mathcal{A}} \{\tilde{X}_{t,\mathcal{B}} = x\}$, which entails

$$|\{X_{t,\mathcal{A}} = x\}| = \sum_{\mathcal{B} \succcurlyeq \mathcal{A}} |\{\tilde{X}_{t,\mathcal{B}} = x\}|.$$

By (2.22), the left-hand side equals $(\bar{R}_{\mathcal{A}}(Z_t))(x)$. Due to the Möbius inversion principle (applied backward), $|\{\tilde{X}_{t,\mathcal{B}} = x\}|$ on the right-hand side must equal $(\bar{H}_{\mathcal{B}}(Z_t))(x)$, as claimed. \square

Lemma 1. *For $\mathcal{A} \in \mathbb{P}(S)$ with $|\mathcal{A}| = m \leq N$ and $z \in E$, $\bar{H}_{\mathcal{A}}(z)$ is a positive measure with*

$$\|\bar{H}_{\mathcal{A}}(z)\| = N(N-1) \cdots (N-m+1) > 0.$$

Proof. Since, under the given assumptions, $(\bar{H}_{\mathcal{A}}(z))(x) = |\{\tilde{X}_{t,\mathcal{A}} = x \mid Z_t = z\}| \geq 0$ for all x by Proposition 3, it is a positive measure, and its norm can be evaluated via

$$\|\bar{H}_{\mathcal{A}}(z)\| = \sum_{x \in \mathbb{X}} |\{\tilde{X}_{t,\mathcal{A}} = x \mid Z_t = z\}|.$$

By means of (2.24), this equals the number of possibilities of how to choose m labelled individuals out of N ones *without* replacement, where the order is respected; this is $N(N-1) \cdots (N-m+1)$, which is positive since $m \leq N$. \square

Under the assumptions of Proposition 3, we can therefore define the normalised version of $\bar{H}_{\mathcal{A}}(z)$ as

$$H_{\mathcal{A}}(z) := \frac{\bar{H}_{\mathcal{A}}(z)}{\|\bar{H}_{\mathcal{A}}(z)\|} = \frac{(N-m)!}{N!} \bar{H}_{\mathcal{A}}(z). \quad (2.25)$$

$H_{\mathcal{A}}(z)$ is the type distribution that results when a sequence is created by taking the letters

for the blocks as encoded by \mathcal{A} from individuals drawn uniformly and *without replacement* from the population z , hence

$$(H_{\mathcal{A}}(z))(x) = \mathbf{P}[\tilde{X}_{t,\mathcal{A}} = x \mid Z_t = z].$$

$H_{\mathcal{A}}$ will later serve as duality function. The situation described here is exactly what happens when a sample is taken in our marginal ancestral recombination process: either the initial sample (according to Σ_0 , from the present population Z_t) or the ancestral one (according to Σ_t , from the initial population Z_0) – hence our name *sampling function*. In this light, Fact 1 expresses counting with replacement in terms of counting without replacement, provided ω is a counting measure.

It is also instructive to express the normalised sampling functions in terms of the normalised recombinators. For $z \in E$ and $|\mathcal{A}| \leq N$, this gives, via (2.21),

$$H_{\mathcal{A}}(z) = \sum_{\mathcal{B} \succ \mathcal{A}} \frac{(N - |\mathcal{A}|)! N^{|\mathcal{B}|}}{N!} \mu(\mathcal{A}, \mathcal{B}) R_{\mathcal{B}}(z). \quad (2.26)$$

Note that $(N - |\mathcal{A}|)! N^{|\mathcal{B}|} / N! = \mathcal{O}(N^{|\mathcal{B}| - |\mathcal{A}|})$. This illustrates how the inclusion of coarser partitions yields higher-order correction terms. The other way round, using (2.21), Fact 1, and (2.25), one gets

$$R_{\mathcal{A}}(z) = \sum_{\mathcal{B} \succ \mathcal{A}} \frac{N!}{N^{|\mathcal{A}|} (N - |\mathcal{B}|)!} H_{\mathcal{B}}(z). \quad (2.27)$$

2.6. Restrictions to subsystems

Recall that we write the restriction of a measure $\omega \in \mathcal{M}_+(\mathbb{X})$ to a subspace $\mathbb{X}_U := \times_{i \in U} \mathbb{X}_i$ of \mathbb{X} as $\pi_U \cdot \omega := \omega \circ \pi_U^{-1}$, which corresponds to marginalisation. Clearly, we can also define recombinators for any non-empty subset $U \subseteq S$ and any partition $\mathcal{A} = \{A_1, \dots, A_m\} \in \mathbb{P}(U)$ as $\bar{R}_{\mathcal{A}}^U(\pi_U \cdot \omega)$, in perfect analogy with $\bar{R}_{\mathcal{A}}^S(\omega)$ for $\mathcal{A} \in \mathbb{P}(S)$, which is $\bar{R}_{\mathcal{A}}(\omega)$; and likewise for $R_{\mathcal{A}}^U$, $\bar{H}_{\mathcal{A}}^U$, and $H_{\mathcal{A}}^U$ (if $\omega \neq 0$). For clarity, we sometimes denote the subsystem by a superscript. However, as in the case of the marginal recombination probabilities, it can be dispensed with since $U = \cup_{j=1}^{|\mathcal{A}|} A_j$ if $\mathcal{A} \in \mathbb{P}(U)$. The interpretation in terms of sampling, as well as Fact 1, carry over.

Restriction of recombinators. Let us collect some basic properties of recombinators:

Fact 2. For $\mathcal{A}, \mathcal{B} \in \mathbb{P}(S)$ and $U, V \subseteq S$ with $S = U \dot{\cup} V$ one has

$$(A) \quad R_{\mathcal{A}} R_{\mathcal{B}} = R_{\mathcal{A} \wedge \mathcal{B}}.$$

$$(B) \quad \pi_U \cdot R_{\mathcal{A}}^S(\omega) = R_{\mathcal{A}|_U}^U(\pi_U \cdot \omega).$$

$$(C) \quad \text{If in addition } \mathcal{A} \preccurlyeq \{U, V\}, \text{ then } \bar{R}_{\mathcal{A}}^S = \bar{R}_{\mathcal{A}|_U}^U \otimes \bar{R}_{\mathcal{A}|_V}^V. \text{ Explicitly, this reads}$$

$$\bar{R}_A^S(\omega) = (\bar{R}_{A|U}^U \otimes \bar{R}_{A|V}^V)(\omega) = (\bar{R}_{A|U}^U(\pi_U \cdot \omega)) \otimes (\bar{R}_{A|V}^V(\pi_V \cdot \omega)).$$

Here and in what follows, we may omit the argument when the meaning is clear.

Proof of Fact 2. Property (A) is Proposition 2 and property (B) is Lemma 1 of [3] (they both remain true in our normalisation). Property (C) is an obvious generalisation of Proposition 2 of [90]. It is easily seen by using first property (A), then (2.20), then (B) and finally (2.20) once more to give

$$\begin{aligned} \bar{R}_A^S(\omega) &= \bar{R}_{\{U,V\}}^S(\bar{R}_A^S(\omega)) = ((\pi_U \cdot \bar{R}_A^S) \otimes (\pi_V \cdot \bar{R}_A^S))(\omega) \\ &= (\bar{R}_{A|U}^U(\pi_U \cdot \omega)) \otimes (\bar{R}_{A|V}^V(\pi_V \cdot \omega)) = (\bar{R}_{A|U}^U \otimes \bar{R}_{A|V}^V)(\omega). \quad \square \end{aligned}$$

Correlations (or linkage disequilibria). Linkage disequilibria (LDE) are used in population genetics to quantify the deviation from independence of allele frequencies at the various sites in a sequence. From three sites onwards, many different notions of linkage disequilibria are available in the literature, see [21, Chap. V.4.2] for an overview.

We will use as LDEs the general correlation functions, which are widely used in statistical physics, see [28] or [71, Chap. 5.1.1]. This results in an explicit formula for multilocus LDEs for an arbitrary number of sites in terms of sums of products of marginal frequencies, see also [9, Appendix] or [50]. As we will see, common definitions for two and three sites coincide with ours.

For any given subset $U \subseteq S$ and $\mathcal{A} \in \mathbb{P}(U)$, we first define *correlation operators* as

$$L_{\mathcal{A}}^U = \sum_{\mathcal{B} \prec \mathcal{A}} \mu(\mathcal{B}, \mathcal{A}) R_{\mathcal{B}}^U. \quad (2.28)$$

Note that the summation is now over all *refinements* of \mathcal{A} , in contrast to our sampling functions, which involve all coarsenings of \mathcal{A} . The restriction to subsystems stems from the fact that one usually considers deviation from independence on small subsets of S .

The $L_{\mathcal{A}}^U$ have a product structure, $L_{\mathcal{A}}^U = \prod_{j=1}^{|\mathcal{A}|} L_1^{A_j}$, which is obvious from (2.28) together with the product structure of the recombinators (Fact 2 (C)) and that of the Möbius function (2.2). Eq. (2.28) has the inverse

$$R_{\mathcal{A}}^U = \sum_{\mathcal{B} \prec \mathcal{A}} L_{\mathcal{B}}^U = \sum_{\mathcal{B} \prec \mathcal{A}} \prod_{j=1}^{|\mathcal{B}|} L_1^{B_j}$$

due to *inversion from below* (see Section 2.2). The latter can be reformulated as

$$L_{\mathcal{A}}^U = R_{\mathcal{A}}^U - \sum_{\mathcal{B} \prec \mathcal{A}} \prod_{j=1}^{|\mathcal{B}|} L_1^{B_j}. \quad (2.29)$$

The case $\mathcal{A} = \mathbf{1}|_U$, $U \subseteq S$, now is of special interest. In line with population-genetics

understanding, we define the *multilocus linkage disequilibrium with respect to the sites in U* by letting L_1^U act on the marginal measure $\pi_U \cdot \omega$, $\omega \in \mathcal{M}_+(\mathbb{X}) \setminus 0$:

$$L_1^U(\pi_U \cdot \omega) = \sum_{\mathcal{A} \in \mathbb{P}(U)} \mu(\mathcal{A}, \mathbf{1}|_U) R_{\mathcal{A}}^U(\pi_U \cdot \omega),$$

cf. (2.28). Note that $L_1^U(\pi_U \cdot \omega)$ is again a measure on $\pi_U(\mathbb{X})$, but no longer positive in general. With the help of (2.29), it can be reformulated as

$$L_1^U(\pi_U \cdot \omega) = R_1^U(\pi_U \cdot \omega) - \sum_{\mathcal{B} \prec \mathbf{1}|_U} \prod_{j=1}^{|\mathcal{B}|} L_1^{B_j}(\pi_{B_j} \cdot \omega),$$

which is Eq. (1) in [50]. Likewise, this alternative formulation of multilocus LDEs agrees with previous ones from [39], [12], and [55] up to $|U| \leq 3$.

Example 2. For $S = \{1, 2, 3, 4\}$ the LDE with respect to the sites in $U = \{2, 4\}$ reads

$$\begin{aligned} L_1^U(\pi_{\{2,4\}} \cdot \omega)(x) &= R_1^U(\pi_{\{2,4\}} \cdot \omega)(x) - R_{\{\{2\}, \{4\}\}}^U(\pi_{\{2,4\}} \cdot \omega)(x) \\ &= \frac{1}{\|\omega\|} \omega(*, x_2, *, x_4) - \frac{1}{\|\omega\|^2} \omega(*, x_2, *, *) \omega(*, *, *, x_4). \end{aligned}$$

Similarly for $U = \{1, 3, 4\}$ we get

$$\begin{aligned} L_1^U(\pi_{\{1,3,4\}} \cdot \omega)(x) &= \frac{1}{\|\omega\|} \omega(x_1, *, x_3, x_4) - \frac{1}{\|\omega\|^2} \omega(x_1, *, *, *) \omega(*, *, x_3, x_4) \\ &\quad - \frac{1}{\|\omega\|^2} \omega(x_1, *, x_3, *) \omega(*, *, *, x_4) - \frac{1}{\|\omega\|^2} \omega(x_1, *, *, x_4) \omega(*, *, x_3, *) \\ &\quad + 2 \frac{1}{\|\omega\|^3} \omega(x_1, *, *, *) \omega(*, *, x_3, *) \omega(*, *, *, x_4). \end{aligned}$$

The correlation operators can also be expressed in terms of our sampling operators. Eqns. (2.28) and (2.27), together with a change of the summation order, lead to

$$\begin{aligned} L_{\mathcal{A}}^U &= \sum_{\mathcal{B} \prec \mathcal{A}} \mu(\mathcal{B}, \mathcal{A}) \sum_{\mathcal{C} \succ \mathcal{B}} \frac{N!}{(N - |\mathcal{C}|)! N^{|\mathcal{B}|}} H_{\mathcal{C}}^U \\ &= \sum_{\mathcal{C} \in \mathbb{P}(U)} H_{\mathcal{C}}^U \sum_{\mathcal{B} \prec \mathcal{A} \wedge \mathcal{C}} \frac{N!}{(N - |\mathcal{C}|)! N^{|\mathcal{B}|}} \mu(\mathcal{B}, \mathcal{A}). \end{aligned} \tag{2.30}$$

For a counting measure $z \in E$ and $U \subseteq S$ with $|U| = k \leq 3 \leq N$, Eq. (2.30) yields a particularly nice explicit expression for the LDEs:

$$L_1^U(\pi_U \cdot z) = \frac{N!}{N^k (N - k)!} \sum_{\mathcal{A} \in \mathbb{P}(U)} \mu(\mathcal{A}, \mathbf{1}|_U) H_{\mathcal{A}}^U(\pi_U \cdot z), \tag{2.31}$$

as is easily verified. For larger k , the explicit formula gets more involved.

Let us now consider $L_{\mathcal{A}}^U$ for $\mathcal{A} \in \mathbb{P}(U) \setminus \mathbf{1}|_U$. Due to its product structure, the collection of all $L_{\mathbf{1}}^V(\pi_V \cdot \omega)$, $V \subseteq U$, determines all $L_{\mathcal{A}}^U(\pi_U \cdot \omega)$, $\mathcal{A} \in \mathbb{P}(U)$. This is why, for a deterministic ω , the $L_{\mathcal{A}}^U(\pi_U \cdot \omega)$, $\mathcal{A} \neq \mathbf{1}|_U$, are of no particular interest of their own. This changes, however, when ω is random (like Z_t). For we typically do not know the law of Z_t completely; rather, we have access to the expectation of certain functions of Z_t . More precisely, let φ be the law of Z_t and \mathbf{E}_{φ} denote the expectation with respect to φ (that is, for a function f of Z_t , $\mathbf{E}_{\varphi}[f] = \int f(z) d\varphi(z)$). It is important to note that the product structure of the recombined measure does not carry over to the expectation. That is, for $\mathcal{A} \in \mathbb{P}(U)$, $\mathbf{E}_{\varphi}[R_{\mathcal{A}}^U(\pi_U \cdot Z_t)] \neq R_{\mathcal{A}}^U(\mathbf{E}_{\varphi}[\pi_U \cdot Z_t])$ in general, see the discussion in [5]; this is indeed a subtle point that sometimes goes wrong, as in [80, Eq. (12)] or [17, pp. 471/472]. As a consequence, one also has $\mathbf{E}_{\varphi}[L_{\mathcal{A}}^U(\pi_U \cdot Z_t)] \neq \prod_{i=1}^{|\mathcal{A}|} L_{\mathbf{1}}^{A_i}(\mathbf{E}_{\varphi}[\pi_{A_i} \cdot Z_t])$ in general. In the stochastic case, therefore, it is interesting to consider the $L_{\mathcal{A}}^U$ for $\mathcal{A} \neq \mathbf{1}|_U$ as well. The expectations $\mathbf{E}_{\varphi}[L_{\mathcal{A}}^U(\pi_U \cdot Z_t)]$ contain information on how the mean LDEs in one part of the sequence depend on the mean LDEs in other parts of the sequence. In Section 2.7 we will obtain an ODE system for the $\mathbf{E}_{\varphi}[H_{\mathcal{A}}^S(Z_t)]$, $\mathcal{A} \in \mathbb{P}(S)$, and these translate into $\mathbf{E}_{\varphi}[R_{\mathcal{A}}^S(Z_t)]$ and thus into $\mathbf{E}_{\varphi}[L_{\mathcal{A}}^S(Z_t)]$ via (2.30). Marginalisation can then be used to calculate the corresponding quantities on $U \subset S$, such as $\mathbf{E}_{\varphi}[L_{\mathcal{A}}^U(\pi_U \cdot Z_t)]$ for $\mathcal{A} \in \mathbb{P}(U)$.

Restriction of the sampling function. First, let us note a connection between recombination and sampling that will be important in what follows in Section 2.7.

Lemma 2. *Let $S = U \dot{\cup} V$ for two nonempty subsets $U, V \subseteq S$. For two partitions $\mathcal{A} \in \mathbb{P}(U)$, $\mathcal{B} \in \mathbb{P}(V)$, the recombinator and the sampling operator satisfy*

$$\bar{R}_{\mathcal{A}}^U \otimes \bar{H}_{\mathcal{B}}^V = \sum_{\substack{\mathcal{C} \succcurlyeq \mathcal{A} \cup \mathcal{B} \\ \mathcal{C}|_V = \mathcal{B}}} \bar{H}_{\mathcal{C}}^S.$$

Proof. Using (2.23) followed by Fact 2 (C) and Fact 1 we get

$$\bar{R}_{\mathcal{A}}^U \otimes \bar{H}_{\mathcal{B}}^V = \bar{R}_{\mathcal{A}}^U \otimes \left(\sum_{\mathcal{D} \succcurlyeq \mathcal{B}} \mu(\mathcal{B}, \mathcal{D}) \bar{R}_{\mathcal{D}}^V \right) = \sum_{\mathcal{D} \succcurlyeq \mathcal{B}} \mu(\mathcal{B}, \mathcal{D}) \bar{R}_{\mathcal{D} \cup \mathcal{A}}^S = \sum_{\mathcal{D} \succcurlyeq \mathcal{B}} \mu(\mathcal{B}, \mathcal{D}) \sum_{\mathcal{E} \succcurlyeq \mathcal{D} \cup \mathcal{A}} \bar{H}_{\mathcal{E}}^S.$$

Changing the summation order and applying (2.1) finally leads to

$$\bar{R}_{\mathcal{A}}^U \otimes \bar{H}_{\mathcal{B}}^V = \sum_{\mathcal{C} \succcurlyeq \mathcal{A} \cup \mathcal{B}} \bar{H}_{\mathcal{C}}^S \sum_{\mathcal{B} \preccurlyeq \mathcal{D} \preccurlyeq \mathcal{C}|_V} \mu(\mathcal{B}, \mathcal{D}) = \sum_{\substack{\mathcal{C} \succcurlyeq \mathcal{A} \cup \mathcal{B} \\ \mathcal{C}|_V = \mathcal{B}}} \bar{H}_{\mathcal{C}}^S. \quad \square$$

Remark 6. *In a perfectly analogous way, one can show*

$$\bar{H}_{\mathcal{A}}^U \otimes \bar{H}_{\mathcal{B}}^V = \sum_{\substack{\mathcal{C} \succcurlyeq \mathcal{A} \cup \mathcal{B} \\ \mathcal{C}|_U = \mathcal{A}, \mathcal{C}|_V = \mathcal{B}}} \bar{H}_{\mathcal{C}}^S.$$

This illustrates once more that, unlike the \bar{R}_A , the \bar{H}_A do not have a product structure; this reflects the dependence inherent to drawing without replacement.

Now, we state two combinatorial facts that will be important for the proof of the following Proposition 4. Therefore, we make use of further combinatorial quantities - in particular *signed* $s(n, k)$ and *unsigned Stirling numbers of first kind* $\left[\begin{smallmatrix} n \\ k \end{smallmatrix} \right] = |s(n, k)|$. Moreover, the notation of falling factorials $[x]_n$ will become handy. Let us recall here only the common property

$$[x]_n = x(x-1) \cdots (x-(n-1)) = \sum_{k=0}^n s(n, k)x^k.$$

See [52] for a summary of the combinatorial properties and recurrence relations for these quantities.

Lemma 3. *Let $N \geq n \geq \tilde{n} \geq 0$. Then*

$$(A) [N - \tilde{n}]_{n-\tilde{n}} = \sum_{k=0}^{n-\tilde{n}} (-1)^{n-\tilde{n}-k} N^k \sum_{l=k}^{n-\tilde{n}} \left[\begin{smallmatrix} n-\tilde{n} \\ l \end{smallmatrix} \right] \binom{l}{k} \tilde{n}^{l-k}.$$

(B) *In addition let U be a set of sites with $\tilde{n} = |U|$ and $\mathcal{C}' \in \mathbb{P}(U)$. Then it holds $\forall S \supseteq U$:*

$$\varsigma(\mathcal{C}') \sum_{l=k}^{n-\tilde{n}} \left[\begin{smallmatrix} n-\tilde{n} \\ l \end{smallmatrix} \right] \binom{l}{k} \tilde{n}^{l-k} = \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} \varsigma(\mathcal{C}) \quad \forall k \in \{0, 1, \dots, n - \tilde{n}\} \quad (2.32)$$

with $|S| = n$ and $\varsigma(\mathcal{C}) := |\mu(\mathbf{0}, \mathcal{C})| = \prod_{i=1}^{|\mathcal{C}|} (|C_i| - 1)!$ for $\mathcal{C} \in \mathbb{P}(S)$.

Proof. Statement (A) follows via direct calculations. Use the definition of signed stirling numbers of first kind and the binomial formula, interchange the obtaining summations and use the definition of unsigned stirling numbers of first kind to obtain:

$$\begin{aligned} [N - \tilde{n}]_{n-\tilde{n}} &= \sum_{l=0}^{n-\tilde{n}} s(n - \tilde{n}, l) (N - \tilde{n})^l = \sum_{l=0}^{n-\tilde{n}} s(n - \tilde{n}, l) \sum_{k=0}^l \binom{l}{k} N^k (-1)^{l-k} \tilde{n}^{l-k} \\ &= \sum_{k=0}^{n-\tilde{n}} N^k \sum_{l=k}^{n-\tilde{n}} \binom{l}{k} s(n - \tilde{n}, l) (-1)^{l-k} \tilde{n}^{l-k} \\ &= \sum_{k=0}^{n-\tilde{n}} N^k (-1)^{n-\tilde{n}-k} \sum_{l=k}^{n-\tilde{n}} \binom{l}{k} s(n - \tilde{n}, l) (-1)^{n-\tilde{n}-l} \tilde{n}^{l-k} \\ &= \sum_{k=0}^{n-\tilde{n}} N^k (-1)^{n-\tilde{n}-k} \sum_{l=k}^{n-\tilde{n}} \binom{l}{k} \left[\begin{smallmatrix} n-\tilde{n} \\ l \end{smallmatrix} \right] \tilde{n}^{l-k}. \end{aligned}$$

Now, we show that (2.32) holds via induction on the values of k ‘from above’. Obviously, the statement holds for $S = U$, so we assume $S \supset U$ from now on.

Base case: Let $k = n - \tilde{n}$. So any additional element $\varepsilon \in S \setminus U$ has to establish a singleton

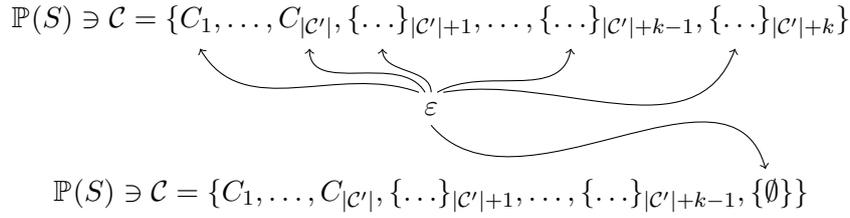


Figure 2.8.: Illustration of the decomposition used in (2.35): a partition $\mathcal{C} \in \mathbb{P}(S)$ with $\mathcal{C}' = \mathcal{C}|_U$ for a given $\mathcal{C}' \in \mathbb{P}(U)$, that contains exactly k additional blocks compared to \mathcal{C}' after the addition of a new element $\varepsilon \notin S$. C_j indicate blocks that contain elements of U whereas $\{\dots\}_k$ indicate those blocks that contain only elements of $S \setminus U$ and thus vanish under the restriction to U .

ε in the partition $\mathcal{C} \in \mathbb{P}(S)$, which yields

$$\varsigma(\mathcal{C}') = \prod_{i=1}^{|\mathcal{C}'|} (|C_i| - 1)! \prod_{i=1}^{n-\tilde{n}} 0! = \varsigma(\mathcal{C}' \cup \bigcup_{\varepsilon \in S \setminus U} \{\{\varepsilon\}\}) = \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + n - \tilde{n}}} \varsigma(\mathcal{C}).$$

Now, let (2.32) hold for $S, S \cup \{\varepsilon\}$, for some $\varepsilon \notin S$, and some k , then

$$\varsigma(\mathcal{C}') \sum_{l=k}^{n-\tilde{n}} \binom{n-\tilde{n}}{l} \binom{l}{k} \tilde{n}^{l-k} = \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} \varsigma(\mathcal{C}), \text{ and} \quad (2.33)$$

$$\varsigma(\mathcal{C}') \sum_{l=k}^{(n+1)-\tilde{n}} \binom{(n+1)-\tilde{n}}{l} \binom{l}{k} \tilde{n}^{l-k} = \sum_{\substack{\mathcal{C} \in \mathbb{P}(S \cup \{\varepsilon\}), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} \varsigma(\mathcal{C}). \quad (2.34)$$

Inductive step $k \rightarrow k-1$: First, consider (2.34) and decompose the sum on the right in the possibilities of adding the element ε to a partitions $\mathcal{C} \in \mathbb{P}(S)$ under the given conditions. Either the new element ε can be filled in one of the blocks of \mathcal{C} , and thus $|\mathcal{C}| = |\mathcal{C}'| + k$, or it establishes a new block, which then has to be a singleton, and thus $|\mathcal{C}| = |\mathcal{C}'| + k - 1$ (see also Figure 2.8). Hence,

$$\begin{aligned} \sum_{\substack{\mathcal{C} \in \mathbb{P}(S \cup \{\varepsilon\}), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} \varsigma(\mathcal{C}) &= \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} \sum_{i=1}^{|\mathcal{C}|} |C_i|! \prod_{j \neq i} (|C_j| - 1)! + \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k - 1}} 0! \varsigma(\mathcal{C}) \quad (2.35) \\ &= \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} n \varsigma(\mathcal{C}) + \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k - 1}} \varsigma(\mathcal{C}). \end{aligned}$$

By using recurrence relations on (2.32) (first for the binomial coefficients, then for the unsigned Stirling number of first kind) followed by some adjustments of the summation

indices, we get

$$\begin{aligned}
& \varsigma(\mathcal{C}') \sum_{l=k-1}^{n-\tilde{n}} \begin{bmatrix} n-\tilde{n} \\ l \end{bmatrix} \binom{l}{k-1} \tilde{n}^{l-(k-1)} \\
&= \varsigma(\mathcal{C}') \left\{ \sum_{l=k-1}^{n-\tilde{n}} \begin{bmatrix} n-\tilde{n} \\ l \end{bmatrix} \binom{l+1}{k} \tilde{n}^{l+1-k} - \tilde{n} \sum_{l=k-1}^{n-\tilde{n}} \begin{bmatrix} n-\tilde{n} \\ l \end{bmatrix} \binom{l}{k} \tilde{n}^{l-k} \right\} \\
&= \varsigma(\mathcal{C}') \left\{ \sum_{l=k-1}^{n-\tilde{n}} \begin{bmatrix} n-\tilde{n}+1 \\ l+1 \end{bmatrix} \binom{l+1}{k} \tilde{n}^{(l+1)-k} \right. \\
&\quad \left. - (n-\tilde{n}) \sum_{l=k-1}^{n-\tilde{n}-1} \begin{bmatrix} n-\tilde{n} \\ l+1 \end{bmatrix} \binom{l+1}{k} \tilde{n}^{(l+1)-k} - \tilde{n} \sum_{l=k}^{n-\tilde{n}} \begin{bmatrix} n-\tilde{n} \\ l \end{bmatrix} \binom{l}{k} \tilde{n}^{l-k} \right\} \\
&= \varsigma(\mathcal{C}') \left\{ \sum_{l=k}^{(n+1)-\tilde{n}} \begin{bmatrix} (n+1)-\tilde{n} \\ l \end{bmatrix} \binom{l}{k} \tilde{n}^{l-k} - n \sum_{l=k}^{n-\tilde{n}} \begin{bmatrix} n-\tilde{n} \\ l \end{bmatrix} \binom{l}{k} \tilde{n}^{l-k} \right\},
\end{aligned}$$

and, finally, using (2.34) followed by (2.35) together with (2.33), gives

$$= \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + (k-1)}} \varsigma(\mathcal{C}) \quad \square$$

Now we want to obtain a result analogous to Fact 2 (B) for H instead of R .

Proposition 4. For $\mathcal{A}, \mathcal{B} \in \mathbb{P}(S)$, a counting measure $z \in E$, and $U \subseteq S$, one has

$$\pi_U H_{\mathcal{A}}^S(z) = H_{\mathcal{A}|_U}^U(\pi_U.z)$$

Proof. For a counting measure $z \in E$, we get

$$H_{\mathcal{A}|_U}^U(\pi_U.z) = \frac{1}{[N]_{|\mathcal{A}|_U}} \sum_{\mathcal{B} \succ \mathcal{A}|_U} N^{|\mathcal{B}|} \mu(\mathcal{A}|_U, \mathcal{B}) R_{\mathcal{B}}^U(\pi_U.z)$$

and

$$\begin{aligned}
\pi_U H_{\mathcal{A}}(z) &= \frac{1}{[N]_{|\mathcal{A}|}} \sum_{\mathcal{B} \succ \mathcal{A}} N^{|\mathcal{B}|} \mu(\mathcal{A}, \mathcal{B}) \pi_U R_{\mathcal{B}}^S(z) = \frac{1}{[N]_{|\mathcal{A}|}} \sum_{\mathcal{B} \succ \mathcal{A}} N^{|\mathcal{B}|} \mu(\mathcal{A}, \mathcal{B}) R_{\mathcal{B}|_U}^U(\pi_U.z) \\
&= \frac{1}{[N]_{|\mathcal{A}|}} \sum_{\mathcal{B} \succ \mathcal{A}|_U} \left(\sum_{\substack{\mathcal{C} \in \mathbb{P}(S) \\ \mathcal{C}|_U = \mathcal{B}}} \mu(\mathcal{A}, \mathcal{C}) N^{|\mathcal{C}|} \right) R_{\mathcal{B}}^U(\pi_U.z).
\end{aligned}$$

By comparing the coefficients, the desired equality holds if and only if

$$\forall \mathcal{B} \succ \mathcal{A}|_U : [N - |\mathcal{A}|_U]_{|\mathcal{A}| - |\mathcal{A}|_U} = \sum_{\substack{\mathcal{C} \in \mathbb{P}(S) \\ \mathcal{C}|_U = \mathcal{B}}} \frac{\mu(\mathcal{A}, \mathcal{C})}{\mu(\mathcal{A}|_U, \mathcal{C}|_U)} N^{|\mathcal{C}| - |\mathcal{C}|_U}, \quad (2.36)$$

see Figure 2.9 for an illustrating example.

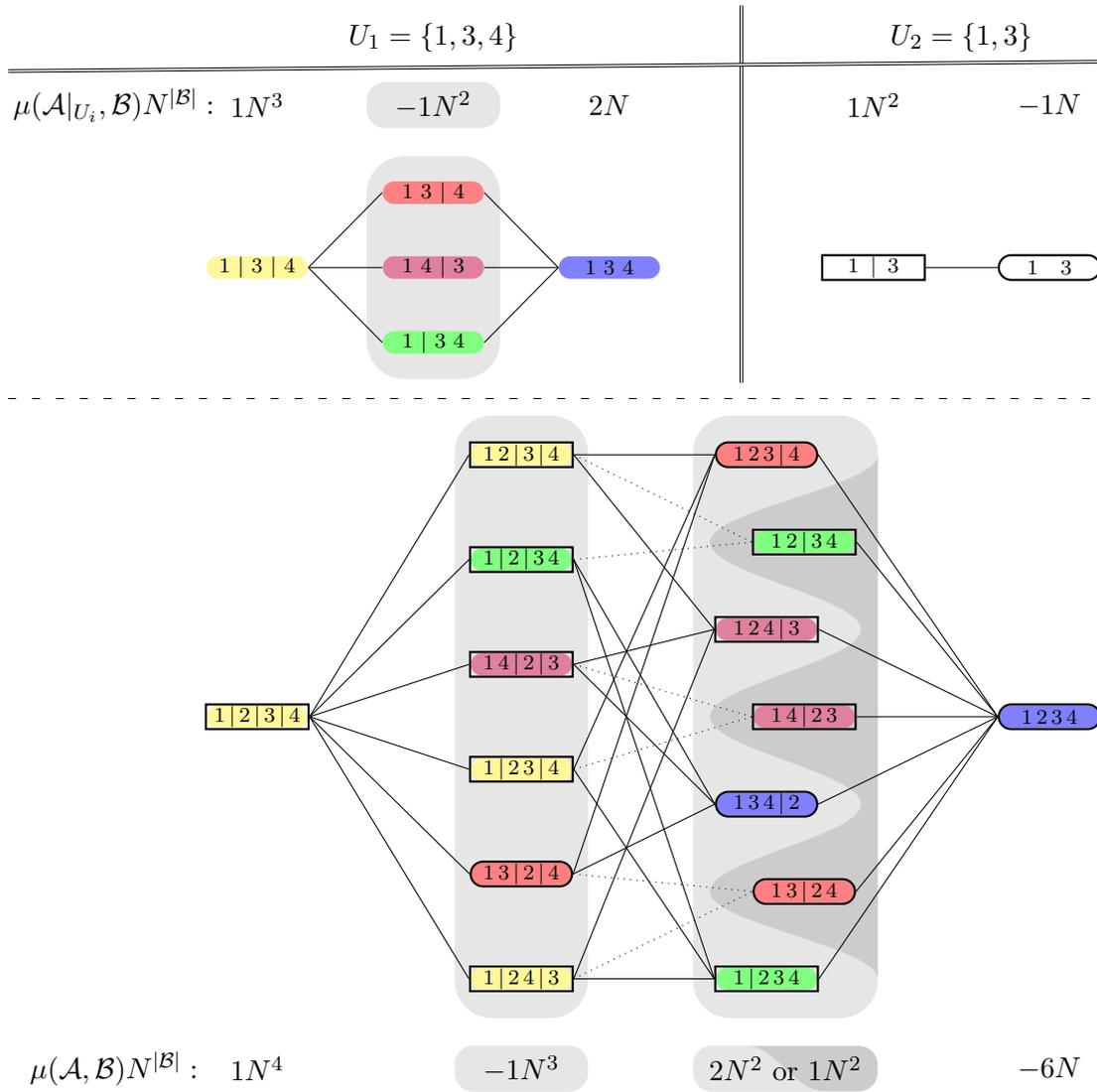


Figure 2.9.: Illustration of the Hasse diagrams of $\{\mathcal{B} \mid \mathcal{B} \succcurlyeq \mathcal{A}\}$ for $\mathcal{A} = \mathbf{0} \in \mathbb{P}(S)$, $S = \{1, 2, 3, 4\}$ and the corresponding restrictions to the subsets $U_1 = \{1, 3, 4\}$ and $U_2 = \{1, 3\}$ together with the values $\mu(\mathcal{A}, \mathcal{B})N^{|\mathcal{B}|}$, $\mu(\mathcal{A}|_{U_i}, \mathcal{B})N^{|\mathcal{B}|}$. For the sake of clarity, we omit the brackets and comma in the partition's representation and separate only the blocks by $|$. The different linestyles and shadings in the lower diagram shall help to differ between the two different classes (depending on $\mu(\mathcal{A}, \mathcal{B})$) of those \mathcal{B} . The association of the partitions $\mathcal{C} \in \mathbb{P}(S)$ to those $\mathcal{B} \succcurlyeq \mathcal{A}|_{U_i}$ to be done in (2.36) is indicated either by different colors ($i = 1$) or boxes ($i = 2$), respectively. We see that, for a given $U \in \{U_1, U_2\}$ and $\mathcal{B} \succcurlyeq \mathcal{A}|_U$, the sum of $\mu(\mathcal{A}, \mathcal{C})N^{|\mathcal{C}|}$ over all $\mathcal{C} \in \mathbb{P}(S)$ with $\mathcal{C}|_U = \mathcal{B} \succcurlyeq \mathcal{A}|_U$ is exactly $\mu(\mathcal{A}|_U, \mathcal{B})N^{|\mathcal{B}|} [N - |\mathcal{A}|_U]_{|\mathcal{A}| - |\mathcal{A}|_U}$, as claimed in the proof of Proposition 4. For example with U_2 and $\mathcal{B} = \{\{1, 3\}\}$, this is $(-1)N(N - 2)(N - 3)$.

Furthermore, $\mu(\mathcal{A}, \mathcal{C})$ implies $\mathcal{C} \succcurlyeq \mathcal{A}$. Insofar, each of the blocks in \mathcal{A} and $\mathcal{A}|_U$, respectively, can be regarded as singletons (in the corresponding system) such that an equivalent reformulation is: if $U \subseteq S$, $|S| = n$, $|U| = \tilde{n}$, then for all $\mathcal{C}' \in \mathbb{P}(U)$, it holds that

$$\begin{aligned} [N - \tilde{n}]_{n-\tilde{n}} &= \sum_{\substack{\mathcal{C} \in \mathbb{P}(S) \\ \mathcal{C}|_U = \mathcal{C}'}} \frac{\mu(\mathbf{0}, \mathcal{C})}{\mu(\mathbf{0}, \mathcal{C}')} N^{|\mathcal{C}| - |\mathcal{C}'|} = \sum_{\substack{\mathcal{C} \in \mathbb{P}(S) \\ \mathcal{C}|_U = \mathcal{C}'}} N^{|\mathcal{C}| - |\mathcal{C}'|} (-1)^{n - \tilde{n} - (|\mathcal{C}| - |\mathcal{C}'|)} \frac{\varsigma(\mathcal{C})}{\varsigma(\mathcal{C}')} \\ \Leftrightarrow \varsigma(\mathcal{C}') [N - \tilde{n}]_{n-\tilde{n}} &= \sum_{k=0}^{n-\tilde{n}} (-1)^{n-\tilde{n}-k} N^k \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} \varsigma(\mathcal{C}). \end{aligned}$$

The application of Lemma 3 (A) leads to the comparison of coefficients for $(-1)^{n-\tilde{n}-k} N^k$

$$\Leftrightarrow \varsigma(\mathcal{C}') \sum_{l=k}^{n-\tilde{n}} \begin{bmatrix} n-\tilde{n} \\ l \end{bmatrix} \binom{l}{k} \tilde{n}^{l-k} = \sum_{\substack{\mathcal{C} \in \mathbb{P}(S), \mathcal{C}|_U = \mathcal{C}' \\ |\mathcal{C}| = |\mathcal{C}'| + k}} \varsigma(\mathcal{C})$$

and (2.32) finishes the proof. \square

Restriction of the partitioning process. In this paragraph we are going to discover that the partitioning process, given by (2.11), can also be restricted to a subsystem $U \subset S$ - or, that it is a *lumpable* process; see [63, §6.3] for the notion of lumpability. This property is auxiliary to show the marginal consistency of the dynamic of the duality results in Section 2.7.

For the sake of identifiability, we use the notation $\underline{\Sigma}_t, \underline{\mathcal{A}}, \underline{A}, \underline{m}$ here to emphasise that a process, partition, set, and size belongs to the subsystem U , respectively (its counterparts in the finer system S will be denoted without underscore, as before). The statement is:

Proposition 5. *The partitioning process Σ_t with state space $\mathbb{P}(S)$ and generator $\Theta_{\mathcal{A}\mathcal{B}}$, given by (2.11) and (2.14), is lumpable, i.e. for any given subset of sites $U \subset S$ the corresponding partitioning process $\underline{\Sigma}_t$ with generator $\Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}}$ and state space $\mathbb{P}(U)$ can be constructed from Σ_t by marginalisation of its transition rates. Namely, for any given $\mathcal{A} \in \mathbb{P}(S)$ with $\mathcal{A}|_U = \underline{\mathcal{A}} \in \mathbb{P}(U)$, it holds*

$$\Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \sum_{\substack{\mathcal{B} \in \mathbb{P}(S) \\ \mathcal{B}|_U = \underline{\mathcal{B}}}} \Theta_{\mathcal{A}\mathcal{B}} \quad \forall \underline{\mathcal{B}} \in \mathbb{P}(U). \quad (2.37)$$

Proof. We proceed in two main steps: first, we show that every non-zero/non-self transition in S yields a valid non-zero (self-)transition in $U \subset S$ by restriction. Then, we change the perspective and elaborate that for any given pair $\underline{\mathcal{A}}, \underline{\mathcal{B}} \in \mathbb{P}(U)$ the relation (2.37) holds for all $\mathcal{A} \in \mathbb{P}(S)$ with $\mathcal{A}|_U = \underline{\mathcal{A}}$, cf. [63, Thm. 6.3.2], and thus the transition rates in U given by (2.11) and (2.14) are consistently obtained by restricting the generator of S to U via marginalisation of the transition probabilities. It is sufficient to show this by separate

explorations of the two possible nondiagonal transition types and the self-transition for the case $S = U \cup \{\varepsilon\}$. The case $U' \subset S$ with $|U'| < |S| - 1$ follows inductively.

Further, we assume that the relevant indices j, k participated in an examined transition shall be chosen appropriately, such that $\underline{A}_j \subseteq A_j$ ($\underline{A}_k \subseteq A_k$). Example 3 after the end of this proof gives some concrete transitions that illustrate the following discussion.

For the first step, we distinct the cases:

Pure coalescence in $\mathbb{P}(S)$, i.e. $\mathcal{A} \rightarrow \mathcal{B} = \mathcal{A}_{M \setminus \{j,k\}} \cup \underline{A}_{\{j,k\}}$ for some $j, k \in M$:

If $\{\varepsilon\}$ is a singleton in \mathcal{A} , then

$$\mathcal{B}|_U = \begin{cases} \mathcal{A}|_U = \mathcal{A}, & \text{if } \{\varepsilon\} \in \{A_j, A_k\}, \\ \mathcal{A}_{M \setminus \{j,k\}}|_U \cup \underline{A}_{\{j,k\}} = \underline{\mathcal{A}}_{M \setminus \{j,k\}} \cup \underline{A}_{\{j,k\}}, & \text{otherwise.} \end{cases}$$

Else, i.e. $\{\varepsilon\} \subsetneq A_i$, $i \in M$, every pure coalescence transition in $\mathbb{P}(S)$ is associated with a valid pure coalescence transition in $\mathbb{P}(U)$, since ε behaves like a free-rider in this case.

Split & coalescence in $\mathbb{P}(S)$, i.e. $\mathcal{B}|_{A_j} = \mathcal{J}$, $\mathcal{B}|_{A_{M \setminus j}} = \mathcal{A}_{M \setminus j}$, for some $j \in M$, $\mathcal{J} \in \mathbb{O}_2(A_j)$:

If $\{\varepsilon\} \subseteq A_\ell$, $\ell \neq j$ the transition $\mathcal{A} \rightarrow \mathcal{B} \in \mathbb{P}(S)$ is obviously associated with the same type of transition in $\mathbb{P}(U)$ by $\mathcal{A}|_U = \mathcal{A} \rightarrow \mathcal{B} = \mathcal{B}|_U$ since ε acts like a free-rider again.

Else, i.e. $\{\varepsilon\} \subsetneq A_j$ ($\{\varepsilon\} = A_j$ belongs to the pure coalescence discussion before), we have to be aware of two subcases:

- Let $\varepsilon \in \{\min A_j, \max A_j\}$ and then w.l.o.g. $\{\varepsilon\} = A_{j_1}$ w.r.t. $\mathcal{J} = \{A_{j_1}, A_{j_2}\}$, so the restriction to U yields either a valid coalescence event $\underline{\mathcal{A}} \rightarrow \underline{\mathcal{B}} = \underline{\mathcal{A}}_{M \setminus \{j,k\}} \cup \underline{A}_{\{j,k\}}$, if A_{j_2} merges, or a self transition $\underline{\mathcal{A}} \rightarrow \underline{\mathcal{A}}$, otherwise.
- If $\min A_j < \varepsilon < \max A_j$, ε acts like a free-rider of either A_{j_1} or A_{j_2} and thus, again, the restriction to U yields a valid non-zero / non-diagonal transition in $\mathbb{P}(U)$.

Since we have seen that every transition in $\mathbb{P}(S)$ has its counterpart in $\mathbb{P}(U)$, we have to show that the marginalised transition probabilities are consistent. Therefore, let $\underline{\mathcal{A}} \rightarrow \underline{\mathcal{B}} \in \mathbb{P}(U)$ be a given non-zero / non-self transition together with a partition $\mathcal{A} \in \mathbb{P}(S)$ s.t. $\mathcal{A}|_U = \underline{\mathcal{A}}$. We have to identify all $\mathcal{B} \in \mathbb{P}(S)$ with $\Theta_{\mathcal{A}\mathcal{B}} \neq 0$ s.t. $\mathcal{B}|_U = \underline{\mathcal{B}}$ and check that (2.37) holds.

Pure coalescence in $\mathbb{P}(U)$, i.e. let $\underline{\mathcal{A}} \rightarrow \underline{\mathcal{B}} = \underline{\mathcal{A}}_{M \setminus \{j,k\}} \cup \underline{A}_{\{j,k\}}$ for some $j, k \in M$:

1. If $\varepsilon \in A_\ell$ for some $\ell \in M \setminus \{j, k\}$ this site does not participate in a relevant coalescence transition in $\mathbb{P}(S)$ and thus $\underline{\mathcal{A}} \rightarrow \underline{\mathcal{B}}$ is represented by a unique $\mathcal{B} \in \mathbb{P}(S)$ with $A_j, A_k \rightarrow A_{\{j,k\}} \in \mathcal{B}$ and $\underline{A}_j = A_j$ as well as $\underline{A}_k = A_k$, so $\Theta_{\mathcal{A}\mathcal{B}} = \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}}$.
2. Else, i.e. $\varepsilon \in A_{\{j,k\}}$ and then w.l.o.g. $\varepsilon \in A_j$, we consider all possible transitions $\mathcal{A} \rightarrow \mathcal{B}$ in $\mathbb{P}(S)$ that are pure coalescence events or yield such in $\mathbb{P}(U)$. Recall that $\{\varepsilon\} \subsetneq A_j$ is the only case of interest, since otherwise a coalescence transition $\mathcal{A} \rightarrow \mathcal{B}$ in $\mathbb{P}(S)$ is a self-transitions in $\mathbb{P}(U)$ (cf. discussion of (pure) coarsening in $\mathbb{P}(S)$ before).

- a) If $\min A_j < \varepsilon < \max A_j$, then $r_{\mathbf{1}}^{A_j} = r_{\mathbf{1}}^{A_j}$ holds and again we have only one $\mathcal{B} \in \mathbb{P}(S)$ that yields a suitable transition, so $\Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}}$.
- b) If $\varepsilon \in \{\min A_j, \max A_j\}$, $|A_j| > 1$ exactly $m - 1$ split & coalescence transitions in $\mathbb{P}(S)$ (either ε joins $A_\ell, j \neq \ell \neq k$ or becomes a singleton) together with the pure coalescence transition $\mathcal{A} \rightarrow \mathcal{B} = \mathcal{A}_{M \setminus \{j,k\}} \cup A_{\{j,k\}}$ yield the same transition in $\mathbb{P}(U)$ by restriction with

$$\begin{aligned} \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} &= (m-2) \frac{(N-(m-1))!}{N^2(N-(m-1))!} r_{\{A_j, \{\varepsilon\}\}}^{A_j \cup \{\varepsilon\}} + \frac{(N-(m-1))!}{N^2(N-m)!} r_{\{A_j, \{\varepsilon\}\}}^{A_j \cup \{\varepsilon\}} \\ &+ \frac{2}{N^2} + \frac{N-1}{N^2} (r_{\mathbf{1}}^{A_j \cup \{\varepsilon\}} + r_{\mathbf{1}}^{A_k}) = \frac{2}{N^2} + \frac{N-1}{N^2} (r_{\mathbf{1}}^{A_j} + r_{\mathbf{1}}^{A_k}). \end{aligned}$$

Here we have used $r_{\mathbf{1}}^{A_j} = r_{\mathbf{1}}^{A_j \cup \{\varepsilon\}} + r_{\{A_j, \{\varepsilon\}\}}^{A_j \cup \{\varepsilon\}}$ and $r_{\mathbf{1}}^{A_k} = r_{\mathbf{1}}^{A_k}$.

Split & coalescence in $\mathbb{P}(U)$, i.e. $\underline{\mathcal{B}}|_{A_j} = \underline{\mathcal{J}}, \underline{\mathcal{B}}|_{A_{M \setminus j}} = \underline{\mathcal{A}}_{M \setminus j}$, f.s. $j \in M, \underline{\mathcal{J}} \in \mathbb{O}_2(A_j)$:

1. Let $\varepsilon \notin A_j \Rightarrow r_{\underline{\mathcal{J}}}^{A_j} = r_{\mathcal{J}}^{A_j}$ for $\underline{\mathcal{J}} = \mathcal{J} = \{A_{j_1}, A_{j_2}\}$ (w.l.o.g. $\max A_{j_1} < \min A_{j_2}$). Then:

- a) Let $\ell \in M$ with $\{\varepsilon\} \subsetneq A_\ell, \ell \neq j$, then $|\underline{\mathcal{B}}| = |\mathcal{B}|$ and $\underline{m} = m$ for a unique $\mathcal{B} \in \mathbb{P}(S)$ with $\mathcal{B}|_U = \underline{\mathcal{B}} \in \mathbb{P}(U)$ and thus $\Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}}$.
- b) Else, i.e. $A_\ell = \{\varepsilon\}, \ell \neq j$, and thus $|\underline{\mathcal{A}}| = m - 1$ as well as $r_{\mathbf{1}}^{A_j} = r_{\mathbf{1}}^{A_j}$, then:

- i. Let $\mathcal{A}' \in \{\{A_{\{j_1, \ell\}}, A_{j_2}\}, \{A_{j_1}, A_{\{j_2, \ell\}}\}, \mathcal{J} \cup \{A_\ell\}\}$, then transitions of the form $\mathcal{A} \rightarrow \mathcal{B} = \mathcal{A}_{M \setminus \{j, \ell\}} \cup \mathcal{A}'$ yield the same $\mathcal{B}|_U = \underline{\mathcal{B}} \in \mathbb{P}(U)$ with $|\underline{\mathcal{B}}| = m$

$$\Rightarrow \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \frac{r_{\underline{\mathcal{J}}}^{A_j}}{N^2} \left[2 \frac{(N-(m-1))!}{(N-m)!} + \frac{(N-(m-1))!}{(N-(m+1))!} \right] = \frac{r_{\underline{\mathcal{J}}}^{A_j}}{N^2} \frac{(N-(\underline{m}-1))!}{(N-|\underline{\mathcal{B}}|)!}.$$

- ii. Let $\mathcal{A}' \in \{\{A_{\{j_1, k, \ell\}}, A_{j_2}\}, \{A_{\{j_1, k\}}, A_{\{j_2, \ell\}}\}, \{A_{\{j_1, k\}}, A_{j_2}, A_\ell\}\}$ for another $k \in M, \ell \neq k \neq j$, then transitions of the form $\mathcal{A} \rightarrow \mathcal{B} = \mathcal{A}_{M \setminus \{j, k, \ell\}} \cup \mathcal{A}'$ yield the same $\mathcal{B}|_U = \underline{\mathcal{B}} \in \mathbb{P}(U)$ with $|\underline{\mathcal{B}}| = m - 1$

$$\Rightarrow \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \frac{r_{\underline{\mathcal{J}}}^{A_j}}{N^2} \left[2 \frac{(N-(m-1))!}{(N-(m-1))!} + \frac{(N-(m-1))!}{(N-m)!} \right] = \frac{r_{\underline{\mathcal{J}}}^{A_j}}{N^2} \frac{(N-(\underline{m}-1))!}{(N-|\underline{\mathcal{B}}|)!}.$$

2. Let $\{\varepsilon\} \subsetneq A_{j_1} (\{\varepsilon\} \subsetneq A_{j_2}) \Rightarrow |\underline{\mathcal{B}}| = |\mathcal{B}|$ and $\underline{m} = m$. (Note here, again, that the case $\{\varepsilon\} = A_{j_1} (\{\varepsilon\} = A_{j_2})$ does not lead to a split & coalescence transitions in $\mathbb{P}(U)$, as already discussed in the first step of this proof).

- a) With $\min A_{j_1} \leq \varepsilon < \max A_{j_1} (\min A_{j_2} < \varepsilon \leq \max A_{j_2})$, then $r_{\underline{\mathcal{J}}}^{A_j} = r_{\mathcal{J}}^{A_j}$ holds and a unique $\mathcal{B} \in \mathbb{P}(S)$ exists with $\mathcal{B}|_U = \underline{\mathcal{B}} \in \mathbb{P}(U)$, which yields $\Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}}$.
- b) Now, let either $\varepsilon = \max A_{j_1}$, such that $A_{j_1} = \underline{A}_{j_1} \cup \{\varepsilon\}, A_{j_2} = \underline{A}_{j_2}$, or $\varepsilon = \min A_{j_2}$, such that $A_{j_1} = \underline{A}_{j_1}, A_{j_2} = \{\varepsilon\} \cup \underline{A}_{j_2}$. Both cases yield the same

$$\begin{aligned} \mathcal{B}|_U = \underline{\mathcal{B}} \in \mathbb{P}(U) \text{ with } \underline{\mathcal{B}}|_{A_j} = \underline{\mathcal{J}} = \{A_{j_1}, A_{j_2}\} \text{ (split either before or after } \varepsilon) \\ \Rightarrow \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \frac{(N - (m - 1))!}{N^2(N - |\underline{\mathcal{B}}|)!} \left[r_{\{A_{j_1} \cup \{\varepsilon\}, A_{j_2}\}}^{A_j} + r_{\{A_{j_1}, \{\varepsilon\} \cup A_{j_2}\}}^{A_j} \right] = \frac{(N - (\underline{m} - 1))!}{N^2(N - |\underline{\mathcal{B}}|)!} r_{\underline{\mathcal{J}}}^{A_j} \end{aligned}$$

Self-transition in $\mathbb{P}(U)$, i.e. $\mathcal{A}|_U = \underline{\mathcal{A}} = \underline{\mathcal{B}}$:

We have to distinct three subcases here, depending on the location of ε in \mathcal{A} :

1. If $\min A_j < \varepsilon < \max A_j$ for some $j \in M \Rightarrow$ no non-self-transition in $\mathbb{P}(S)$ can make $\{\varepsilon\}$ act as a free-rider, so $r_{\mathbf{1}}^{A_j} = r_{\mathbf{1}}^{A_j}$ and thus $\Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}} = \Theta_{\underline{\mathcal{A}}\underline{\mathcal{B}}}$.
2. If $\exists j \in M$ s.t. $\varepsilon \in \{\min A_j, \max A_j\}$, $|A_j| > 1$, then in total $m - 1$ split & coalescence transitions with $\underline{m} = m$ yield a $\underline{\mathcal{B}} \neq \underline{\mathcal{A}} \in \mathbb{P}(S)$ with $\underline{\mathcal{B}}|_U = \underline{\mathcal{A}}$, (2.37) holds for $\underline{\mathcal{B}} = \underline{\mathcal{A}} = \underline{\mathcal{A}}|_U$ due to $r_{\text{tot}}(\underline{\mathcal{A}}) = r_{\text{tot}}(\underline{\mathcal{A}}) + r_{\{\{\varepsilon\}, A_j\}}^{A_j}$, and

$$\frac{(N - (\underline{m} - 1))!}{N^2(N - (\underline{m} + 1))!} + (m - 1) \frac{(N - (\underline{m} - 1))!}{N^2(N - \underline{m})!} = \frac{(N - 1)(N - (\underline{m} - 1))}{N^2}.$$

3. Otherwise, i.e. $\exists j \in M$ with $\{\varepsilon\} = A_j$, we have $A_k = \underline{A}_k \forall k \in M \setminus j$ and $\underline{m} = m - 1$. Thus $r_{\text{tot}}(\underline{\mathcal{A}}) = r_{\text{tot}}(\underline{\mathcal{A}}) = \sum_{k \in M \setminus j} \sum_{\mathcal{J} \in \mathcal{O}_2(A_k)} r_{\mathcal{J}}^{A_k}$ and $m - 1$ coalescence transitions $\underline{\mathcal{A}} \rightarrow \underline{\mathcal{B}}$ exist in $\mathbb{P}(S)$ with $\underline{\mathcal{B}}|_U = \underline{\mathcal{A}}$

$$\begin{aligned} \Rightarrow \Theta_{\underline{\mathcal{A}}\underline{\mathcal{A}}} &= - \left(\frac{(\underline{m} + 1)((\underline{m} - 1) + 1)}{N} + \frac{(N - 1)(N - (\underline{m} - 1) - 1)}{N^2} r_{\text{tot}}(\underline{\mathcal{A}}) \right) \\ &\quad + \sum_{k \in M \setminus j} \left(\frac{2}{N^2} + \frac{N - 1}{N^2} (1 + r_{\mathbf{1}}^{A_k}) \right) \\ &= - \left(\frac{\underline{m}(\underline{m} - 1)}{N} + \frac{(N - 1)(N - (\underline{m} - 1))}{N^2} r_{\text{tot}}(\underline{\mathcal{A}}) \right). \quad \square \end{aligned}$$

Example 3. Let us give some short examples that shall help to trace the proof of Proposition 5 with $\varepsilon = 3 \in S = \{1, 2, 3, 4, 5, 6\}$. The first arrow indicates the transition in $\mathbb{P}(S)$ and the second arrow indicates the restriction to $U = S \setminus \varepsilon$.

(Pure) coalescence in $\mathbb{P}(U)$, i.e. $\underline{\mathcal{A}} \rightarrow \underline{\mathcal{B}} = \underline{\mathcal{A}}_{M \setminus \{j, k\}} \cup \underline{\mathcal{A}}_{\{j, k\}}$ for some $j, k \in \underline{M}$:

1. $\{\{1, 2\}, \{3, 4\}, \{5, 6\}\} \rightarrow \{\{1, 2, 5, 6\}, \{3, 4\}\} \rightsquigarrow \{\{1, 2, 5, 6\}, \{4\}\}$
2. (Otherwise) $\{\{1, 2\}, \{3\}, \{4\}, \{5, 6\}\} \rightarrow \{\{1, 2, 3\}, \{4\}, \{5, 6\}\} \rightsquigarrow \{\{1, 2\}, \{4\}, \{5, 6\}\}$
 - a) $\{\{1, 3, 6\}, \{2, 4\}, \{5\}\} \rightarrow \{\{1, 2, 3, 4, 6\}, \{5\}\} \rightsquigarrow \{\{1, 2, 4, 6\}, \{5\}\}$
 - b) $\{\{1 : 3\}, \{2, 5\}, \{4\}, \{6\}\} \rightarrow \left| \begin{array}{l} \{\{1, 2, 5\}, \{3, 4\}, \{6\}\} \\ \{\{1, 2, 5\}, \{4\}, \{3, 6\}\} \\ \{\{1, 2, 5\}, \{3\}, \{4\}, \{6\}\} \\ \{\{1, 2, 3, 5\}, \{4\}, \{6\}\} \end{array} \right| \rightsquigarrow \{\{1, 2, 5\}, \{4\}, \{6\}\}$

Split & coalescence in $\mathbb{P}(U)$, i.e. $\underline{\mathcal{B}}|_{A_j} = \underline{\mathcal{J}}, \underline{\mathcal{B}}|_{A_{M \setminus j}} = \underline{\mathcal{A}}_{M \setminus j}$, for some $j \in \underline{M}$, $\underline{\mathcal{J}} \in \mathcal{O}_2(\underline{A}_j)$:

1. a) $\{\{1 : 4\}, \{2, 5\}, \{3, 6\}\} \rightarrow \{\{1, 3, 6\}, \{2, 4, 5\}\} \rightsquigarrow \{\{1, 6\}, \{2, 4, 5\}\}$

$$\begin{aligned}
b) \quad i. \quad & \{\{1 \dot{:} 4\}, \{2, 5\}, \{3\}, \{6\}\} \rightarrow \left| \begin{array}{l} \{\{1\}, \{4\}, \{2, 5\}, \{3\}, \{6\}\} \\ \{\{1\}, \{3, 4\}, \{2, 5\}, \{6\}\} \\ \{\{1, 3\}, \{4\}, \{2, 5\}, \{6\}\} \end{array} \right| \rightsquigarrow \{\{1\}, \{4\}, \{2, 5\}, \{6\}\} \\
ii. \quad & \{\{1 \dot{:} 4\}, \{2, 5\}, \{3\}, \{6\}\} \rightarrow \left| \begin{array}{l} \{\{1, 2, 5\}, \{4\}, \{3, 6\}\} \\ \{\{1, 2, 5\}, \{4, 3\}, \{6\}\} \end{array} \right| \rightsquigarrow \{\{1, 2, 5\}, \{4\}, \{6\}\} \\
2. \quad a) \quad & \{\{1, 3, 4 \dot{:} 6\}, \{2, 5\}\} \rightarrow \{\{1, 3, 4\}, \{2, 5, 6\}\} \rightsquigarrow \{\{1, 4\}, \{2, 5, 6\}\} \\
b) \quad & \{\{1, 3 \dot{:} 4\}, \{2, 5\}, \{6\}\} \rightarrow \left| \begin{array}{l} \{\{1, 2, 5\}, \{3, 4, 6\}\} \\ \{\{1, 2, 3, 5\}, \{4, 6\}\} \end{array} \right| \rightsquigarrow \{\{1, 2, 5\}, \{4, 6\}\}
\end{aligned}$$

Self-transition in $\mathbb{P}(U)$, i.e. $\mathcal{A}|_U = \underline{\mathcal{A}} = \underline{\mathcal{B}}$:

$$\begin{aligned}
1. \quad & \{\{1, 3, 4\}, \{2\}, \{5, 6\}\} \rightarrow \{\{1, 3, 4\}, \{2\}, \{5, 6\}\} \rightsquigarrow \{\{1, 4\}, \{2\}, \{5, 6\}\} \\
2. \quad & \{\{12\}, \{3\}, \{4, 5\}, \{6\}\} \rightarrow \left| \begin{array}{l} \{\{1, 2, 3\}, \{4, 5\}, \{6\}\} \\ \{\{1, 2\}, \{3, 4, 5\}, \{6\}\} \\ \{\{1, 2\}, \{4, 5\}, \{3, 6\}\} \\ \{\{1, 2\}, \{3\}, \{4, 5\}, \{6\}\} \end{array} \right| \rightsquigarrow \{\{1, 2\}, \{4, 5\}, \{6\}\} \\
3. \quad & \{\{1, 3\}, \{2, 4\}, \{5\}, \{6\}\} \rightarrow \left| \begin{array}{l} \{\{1\}, \{2, 3, 4\}, \{5\}, \{6\}\} \\ \{\{1\}, \{2, 4\}, \{3, 5\}, \{6\}\} \\ \{\{1\}, \{2, 4\}, \{5\}, \{3, 6\}\} \\ \{\{1\}, \{3\}, \{2, 4\}, \{5, 6\}\} \\ \{\{1, 3\}, \{2, 4\}, \{5\}, \{6\}\} \end{array} \right| \rightsquigarrow \{\{1\}, \{2, 4\}, \{5\}, \{6\}\}
\end{aligned}$$

2.7. Duality

Duality is a powerful tool to obtain information about one process by studying another, the dual process. The latter may, in an optimal case, have a much smaller state space than the original one. Duality results are essential in interacting particle systems in physics and in population genetics. They are often related to time reversal. The most famous example in population genetics is arguably the moment duality between the Wright-Fisher diffusion forward in time and the block counting process of Kingman's coalescent backward in time (see [25], [72]). In [68] this result is extended by incorporating recombination into the two-locus, two-allele case. Those results are based on the original version of the ARG and thus on the diffusion limit.

We will briefly explain the general duality concept and then prove that our processes $\{Z_t\}_{t \geq 0}$ and $\{\Sigma_t\}_{t \geq 0}$ are duals of each other. For the general principle, let $X = \{X_t\}_{t \geq 0}$ and $Y = \{Y_t\}_{t \geq 0}$ be two Markov processes with state spaces E and F . Define by $M(E \times F)_b$ the set of all bounded, measurable functions on $E \times F$. The following definition of duality with respect to a function goes back to [67]; see also the recent review in [59].

Definition 3 (Duality). *The Markov processes X and Y , with laws φ and ψ , respectively, are said to be dual with respect to a function $H \in M(E \times F)_b$ if, for all $x \in E$, $y \in F$, and $t \geq 0$,*

$$\mathbf{E}_\varphi [H(X_t, y) \mid X_0 = x] = \mathbf{E}_\psi [H(x, Y_t) \mid Y_0 = y]. \quad (2.38)$$

If E and F are finite, every function $H \in M(E \times F)_b$ may be represented by a matrix with bounded entries $H(v, w)$, $v \in E$, $w \in F$. If, further, X and Y are time-homogeneous with generator matrices Λ and Θ respectively, the expectations in (2.38) may be written in terms of the corresponding semigroups, i.e.,

$$\begin{aligned}\mathbf{E}_\varphi [H(X_t, y) \mid X_0 = x] &= \sum_{v \in E} (e^{t\Lambda})_{xv} H(v, y), \\ \mathbf{E}_\psi [H(x, Y_t) \mid Y_0 = y] &= \sum_{w \in F} (e^{t\Theta})_{yw} H(x, w).\end{aligned}\tag{2.39}$$

Since the duality equation (2.38) is automatically satisfied at $t = 0$, it is sufficient to check the identity of the derivatives at $t = 0$. That is, Eq. (2.38) holds for all times if and only if

$$\begin{aligned}\frac{d}{dt} \mathbf{E}_\varphi [H(X_t, y) \mid X_0 = x] \Big|_{t=0} &= \sum_{v \in E} \Lambda_{xv} H(v, y) \\ &= \sum_{w \in F} H(x, w) \Theta_{yw} \\ &= \frac{d}{dt} \mathbf{E}_\psi [H(x, Y_t) \mid Y_0 = y] \Big|_{t=0}\end{aligned}\tag{2.40}$$

for all $x \in E$, $y \in F$. As a short-hand of (2.40), one can write $\Lambda H = H\Theta^T$, where T denotes transpose.

We will now present a duality result that justifies our construction of a marginalised sample at present via the partitioning process and sampling from the initial population (cf. Figure 2.4). Indeed, it is not coincidence that we have denoted our sampling functions by $H_{\mathcal{A}}$ and our generators by Λ and Θ .

Theorem 1. *The population process $\{Z_t\}_{t \geq 0}$ and the partitioning process $\{\Sigma_t\}_{t \geq 0}$ with the generators Λ and Θ and resulting laws φ and ψ , respectively, are dual with respect to the sampling function H defined in (2.25). Explicitly,*

$$\mathbf{E}_\varphi [H_{\mathcal{A}}(Z_t) \mid Z_0 = z] = \mathbf{E}_\psi [H_{\Sigma_t}(z) \mid \Sigma_0 = \mathcal{A}]\tag{2.41}$$

for all $\mathcal{A} \in \mathbb{P}(S)$ and $z \in E$.

Before we embark on the proof, let us briefly comment on the meaning of this result.

Remark 7. *Eq. (2.41) is the formal equivalent of the construction in Figure 2.4. To see this, recall the random variables $\tilde{X}_{t, \mathcal{A}}$ from (2.24). With their help, the left-hand side of (2.41) may be reformulated as a probability distribution,*

$$\mathbf{E}_\varphi [H_{\mathcal{A}}(Z_t) \mid Z_0 = z] = \mathbf{E}_\varphi [\mathbf{P}[\tilde{X}_{t, \mathcal{A}} = \cdot] \mid Z_t, Z_0 = z] = \mathbf{P}_\varphi [\tilde{X}_{t, \mathcal{A}} = \cdot \mid Z_0 = z],$$

since the expectation is over all realisations of Z_t . The right-hand side is the probability

distribution considered in [17]. Likewise, the right-hand side of (2.41) is equal to

$$\mathbf{E}_\psi[H_{\Sigma_t}(z) \mid \Sigma_0 = \mathcal{A}] = \mathbf{E}_\psi[\mathbf{P}[\tilde{X}_{0,\Sigma_t} = \cdot \mid \Sigma_t, \Sigma_0 = \mathcal{A}] = \mathbf{P}_\psi[\tilde{X}_{0,\Sigma_t} = \cdot \mid \Sigma_0 = \mathcal{A}],$$

since the expectation is over all realisations of Σ_t . The right-hand side is the distribution of types when sampling from the initial population according to the partition Σ_t , where it is understood that the initial population consists of the types X_0^1, \dots, X_0^N with $\sum_{k=1}^N \delta_{X_0^k} = z$. Recall that time runs forward in Z_t , X_t^k , and $\tilde{X}_{t,\mathcal{A}}$, but backward in Σ_t .

In order to avoid case distinctions in the calculations in the remainder of this section, let us agree on the following conventions concerning (partitions of) empty sets. Namely, we set $\mathcal{A}_\emptyset := \emptyset$, $\bar{H}_\emptyset(\pi_\emptyset \cdot z) = \bar{R}_\emptyset(\pi_\emptyset \cdot z) := \pi_\emptyset \cdot z = \|z\| = N$, and $\mu(\emptyset, \emptyset) := 1$. We now collect some auxiliary results in the following Lemma.

Lemma 4. *Consider a counting measure $z \in E$, a partition $\mathcal{A} \in \mathbb{P}(S)$ with $|\mathcal{A}| = m \leq N$ and corresponding index set $M = \{1, \dots, m\}$, and a partition $\mathcal{B} \in \mathbb{P}(S)$. Then, the following statements hold:*

$$(A) \sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) [\bar{H}_{\mathcal{A}}(z + \delta_x) - \bar{H}_{\mathcal{A}}(z)] = \sum_{j \in M} \left(\bar{H}_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{B}|_{\mathcal{A}_j}} \right)(z).$$

$$(B) \sum_{x \in \mathbb{X}} z(x) [\bar{H}_{\mathcal{A}}(z - \delta_x) - \bar{H}_{\mathcal{A}}(z)] = -m \bar{H}_{\mathcal{A}}(z).$$

Before we prove the lemma, let us give some explanations.

Remark 8. *Note first that, with the above conventions, the right-hand side of identity (A) evaluates to*

$$\sum_{j \in M} \left(\bar{H}_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{B}|_{\mathcal{A}_j}} \right)(z) = N R_{\mathcal{B}}(z)$$

if $\mathcal{A} = \mathbf{1}$.

Let us now provide an interpretation for the statements of the lemma. Evaluating statement (A) for a given type $y \in \mathbb{X}$ yields the equivalent formulation

$$\left(\sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) \bar{H}_{\mathcal{A}}(z + \delta_x) \right)(y) = (\bar{H}_{\mathcal{A}}(z))(y) + \sum_{j \in M} \left(\left(\bar{H}_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{B}|_{\mathcal{A}_j}} \right)(z) \right)(y).$$

Let us read the left-hand side as the expected number of y individuals when drawing the parts of \mathcal{A} without replacement from the population z to which one individual with type distribution $R_{\mathcal{B}}(z)$ has been added. The statement then says that this can be achieved either by drawing all parts of \mathcal{A} from z without replacement, or by drawing all but one of them from z without replacement and the parts of \mathcal{B} induced by the remaining block independently of each other and of all other blocks.

Likewise, evaluating statement (B) for some type $y \in \mathbb{X}$ gives

$$\left(\sum_{x \in \mathbb{X}} \frac{z(x)}{N} \bar{H}_{\mathcal{A}}(z - \delta_x) \right)(y) = \frac{N-m}{N} (\bar{H}_{\mathcal{A}}(z))(y).$$

Let us note in passing that the left-hand side is always well-defined, since $z - \delta_x < 0$ can only occur with $z(x) = 0$, in which case the term vanishes. This left-hand side yields the expected number of y individuals when drawing the parts of \mathcal{A} from the population z after removal of one randomly sampled individual. The statement then tells us that this is the same as first drawing the parts of \mathcal{A} from all of z and then deciding whether none of the m affected individuals has been removed, which is the case with probability $(N - m)/N$.

Proof of Lemma 4. We first observe that

$$\sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) (\pi_U \cdot \delta_x) = \pi_U \cdot \sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) \delta_x = \pi_U \cdot (R_{\mathcal{B}}(z)) = R_{\mathcal{B}|_U}(\pi_U \cdot z) \quad (2.42)$$

by Fact 2. We next evaluate $\sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) [\bar{R}_{\mathcal{A}}(z + \delta_x) - \bar{R}_{\mathcal{A}}(z)]$ by expanding $\bar{R}_{\mathcal{A}}$ to separate the action on z from that on δ_x , summing against $R_{\mathcal{B}}(z)$ (using (2.42)), applying Fact 1 and changing summation:

$$\begin{aligned} & \sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) [\bar{R}_{\mathcal{A}}(z + \delta_x) - \bar{R}_{\mathcal{A}}(z)] \\ &= \sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) \sum_{\emptyset \neq J \subseteq M} \left(\bar{R}_{\mathcal{A}_{M \setminus J}}(\pi_{\mathcal{A}_{M \setminus J}} \cdot z) \right) \otimes (\pi_{\mathcal{A}_J} \cdot \delta_x) \\ &= \sum_{\emptyset \neq J \subseteq M} \left(\bar{R}_{\mathcal{A}_{M \setminus J}} \otimes R_{\mathcal{B}|_{\mathcal{A}_J}} \right)(z) = \sum_{\emptyset \neq J \subseteq M} \sum_{\mathcal{C} \not\ni \mathcal{A}_{M \setminus J}} \left(\bar{H}_{\mathcal{C}} \otimes R_{\mathcal{B}|_{\mathcal{A}_J}} \right)(z) \\ &= \sum_{\mathcal{D} \not\ni \mathcal{A}} \sum_{j=1}^{|\mathcal{D}|} \left(\bar{H}_{\mathcal{D} \setminus D_j} \otimes R_{\mathcal{B}|_{D_j}} \right)(z), \end{aligned}$$

where, in the last step, every \mathcal{A}_J reappears as one D_j . Using this together with (2.23) and (2.1), we obtain

$$\begin{aligned} & \sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) [\bar{H}_{\mathcal{A}}(z + \delta_x) - \bar{H}_{\mathcal{A}}(z)] = \sum_{\mathcal{C} \not\ni \mathcal{A}} \mu(\mathcal{A}, \mathcal{C}) \sum_{x \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) [\bar{R}_{\mathcal{C}}(z + \delta_x) - \bar{R}_{\mathcal{C}}(z)] \\ &= \sum_{\mathcal{C} \not\ni \mathcal{A}} \mu(\mathcal{A}, \mathcal{C}) \sum_{\mathcal{D} \not\ni \mathcal{C}} \sum_{j=1}^{|\mathcal{D}|} \left(\bar{H}_{\mathcal{D} \setminus D_j} \otimes R_{\mathcal{B}|_{D_j}} \right)(z) \\ &= \sum_{\mathcal{D} \not\ni \mathcal{A}} \sum_{j=1}^{|\mathcal{D}|} \left(\bar{H}_{\mathcal{D} \setminus D_j} \otimes R_{\mathcal{B}|_{D_j}} \right)(z) \sum_{\mathcal{A} \preceq \mathcal{C} \preceq \mathcal{D}} \mu(\mathcal{A}, \mathcal{C}) = \sum_{j \in M} \left(\bar{H}_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{B}|_{\mathcal{A}_j}} \right)(z), \end{aligned}$$

which is statement (A). In an analogous way, we can prove statement (B):

$$\begin{aligned} & \sum_{x \in \mathbb{X}} z(x) [\bar{R}_{\mathcal{A}}(z - \delta_x) - \bar{R}_{\mathcal{A}}(z)] \\ &= \sum_{\emptyset \neq J \subseteq M} (-1)^{|J|} \sum_{x \in \mathbb{X}} z(x) \left(\bar{R}_{\mathcal{A}_{M \setminus J}}(\pi_{\mathcal{A}_{M \setminus J}} \cdot z) \right) \otimes \left(\bar{R}_{\mathbf{1}}^{\mathcal{A}_J}(\pi_{\mathcal{A}_J} \cdot \delta_x) \right) \end{aligned}$$

$$\begin{aligned}
&= \sum_{\emptyset \neq J \subseteq M} (-1)^{|J|} \left(\bar{R}_{\mathcal{A}_{M \setminus J}} \otimes \bar{R}_1^{A_J} \right) (z) = \sum_{\emptyset \neq J \subseteq M} (-1)^{|J|} \left(\bar{R}_{\mathcal{A}_{M \setminus J} \cup A_J} \right) (z) \\
&= \sum_{\emptyset \neq J \subseteq M} (-1)^{|J|} \sum_{\mathcal{B} \succ \mathcal{A}_{M \setminus J} \cup A_J} \bar{H}_{\mathcal{B}}(z) = \sum_{\mathcal{C} \succ \mathcal{A}} \bar{H}_{\mathcal{C}}(z) \sum_{j=1}^{|\mathcal{C}|} \sum_{\emptyset \neq K \subseteq C_j} (-1)^{|K|} \\
&= \sum_{\mathcal{C} \succ \mathcal{A}} \bar{H}_{\mathcal{C}}(z) \sum_{j=1}^{|\mathcal{C}|} \left[(1-1)^{|C_j|} - 1 \right] = - \sum_{\mathcal{C} \succ \mathcal{A}} |\mathcal{C}| \bar{H}_{\mathcal{C}}(z),
\end{aligned}$$

where, in the second-last step, every A_J reappears as a C_j . We therefore get

$$\begin{aligned}
\sum_{x \in \mathbb{X}} z(x) [\bar{H}_{\mathcal{A}}(z - \delta_x) - \bar{H}_{\mathcal{A}}(z)] &= \sum_{\mathcal{B} \succ \mathcal{A}} \mu(\mathcal{A}, \mathcal{B}) \sum_{x \in \mathbb{X}} z(x) [\bar{R}_{\mathcal{B}}(z - \delta_x) - \bar{R}_{\mathcal{B}}(z)] \\
&= - \sum_{\mathcal{B} \succ \mathcal{A}} \mu(\mathcal{A}, \mathcal{B}) \sum_{\mathcal{C} \succ \mathcal{B}} |\mathcal{C}| \bar{H}_{\mathcal{C}}(z) = - \sum_{\mathcal{C} \succ \mathcal{A}} |\mathcal{C}| \bar{H}_{\mathcal{C}}(z) \sum_{\mathcal{A} \prec \mathcal{B} \prec \mathcal{C}} \mu(\mathcal{A}, \mathcal{B}) \\
&= -|\mathcal{A}| \bar{H}_{\mathcal{A}}(z),
\end{aligned}$$

as claimed. \square

We can now proceed as follows.

Proof of Theorem 1. We start with the partitioning process. We first note that

$$\sum_{\substack{\mathcal{B} \succ \mathcal{A}_{M \setminus j} \cup \mathcal{J} \\ \mathcal{B}|_{\mathcal{A}_{M \setminus j}} = \mathcal{A}_{M \setminus j}}} \frac{(N - (m - 1))!}{(N - |\mathcal{B}|)!} = N^{|\mathcal{J}|} \quad (2.43)$$

for $j \in M$ and $|\mathcal{J}| \leq 2$. This is easily verified by direct calculation; namely, for $|\mathcal{J}| = 1$, the sum on the left-hand side equals $(N - (m - 1)) + (m - 1) = N$; for $|\mathcal{J}| = 2$, it evaluates to

$$(N - (m - 1))(N - m) + (N - (m - 1))(2m - 1) + (m - 1)^2 = N^2.$$

We now use the formulation of the process via (2.10) and (2.11) in the first step, normalisation and (2.43) in the second, Lemma 2 in the third, and finally another normalisation step to calculate

$$\begin{aligned}
\sum_{\mathcal{B} \in \mathbb{P}(S)} \theta_{\mathcal{A}\mathcal{B}} H_{\mathcal{B}}(z) &= \sum_{j \in M} \sum_{\mathcal{J} \in \mathbb{0}_{\leq 2}(A_j)} \frac{r_{\mathcal{J}}}{N^{|\mathcal{J}|}} \sum_{\substack{\mathcal{B} \succ \mathcal{A}_{M \setminus j} \cup \mathcal{J} \\ \mathcal{B}|_{\mathcal{A}_{M \setminus j}} = \mathcal{A}_{M \setminus j}}} \frac{(N - (m - 1))!}{(N - |\mathcal{B}|)!} (H_{\mathcal{B}} - H_{\mathcal{A}})(z) \\
&= \sum_{j \in M} \sum_{\mathcal{J} \in \mathbb{0}_{\leq 2}(A_j)} \frac{r_{\mathcal{J}}}{N^{|\mathcal{J}|}} \left(\left(\sum_{\substack{\mathcal{B} \succ \mathcal{A}_{M \setminus j} \cup \mathcal{J} \\ \mathcal{B}|_{\mathcal{A}_{M \setminus j}} = \mathcal{A}_{M \setminus j}}} \frac{(N - (m - 1))!}{N!} \bar{H}_{\mathcal{B}} \right) - N^{|\mathcal{J}|} H_{\mathcal{A}} \right) (z)
\end{aligned}$$

$$\begin{aligned}
&= \sum_{j \in M} \sum_{\mathcal{J} \in \mathcal{O}_{\leq 2}(A_j)} \frac{r_{\mathcal{J}}}{N^{|\mathcal{J}|}} \left(\frac{(N - (m - 1))!}{N!} (\bar{H}_{\mathcal{A}_{M \setminus j}} \otimes \bar{R}_{\mathcal{J}}) - N^{|\mathcal{J}|} H_{\mathcal{A}} \right) (z) \\
&= \sum_{j \in M} \sum_{\mathcal{J} \in \mathcal{O}_{\leq 2}(A_j)} r_{\mathcal{J}} (H_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{J}} - H_{\mathcal{A}}) (z). \tag{2.44}
\end{aligned}$$

We now turn to the type distribution process. Here we first evaluate, with Lemma 4 (B):

$$\begin{aligned}
&\sum_{y \in \mathbb{X}} z(y) [\bar{H}_{\mathcal{A}}(z + \delta_x - \delta_y) - \bar{H}_{\mathcal{A}}(z)] \\
&= \sum_{y \in \mathbb{X}} (z + \delta_x)(y) \bar{H}_{\mathcal{A}}((z + \delta_x) - \delta_y) - \sum_{y \in \mathbb{X}} (z + \delta_x)(y) \bar{H}_{\mathcal{A}}(z) \\
&= \sum_{y \in \mathbb{X}} (z + \delta_x)(y) [\bar{H}_{\mathcal{A}}((z + \delta_x) - \delta_y) - \bar{H}_{\mathcal{A}}(z + \delta_x) + \bar{H}_{\mathcal{A}}(z + \delta_x) - \bar{H}_{\mathcal{A}}(z)] \\
&= (N + 1 - m) [\bar{H}_{\mathcal{A}}(z + \delta_x) - \bar{H}_{\mathcal{A}}(z)] - m \bar{H}_{\mathcal{A}}(z).
\end{aligned}$$

From this, we obtain via summation against $R_{\mathcal{B}}(z)$ and use of Lemma 4 (A) that

$$\begin{aligned}
&\sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) z(y) [\bar{H}_{\mathcal{A}}(z + \delta_x - \delta_y) - \bar{H}_{\mathcal{A}}(z)] \\
&= (N + 1 - m) \sum_{j \in M} (\bar{H}_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{B}|_{A_j}})(z) - m \bar{H}_{\mathcal{A}}(z). \tag{2.45}
\end{aligned}$$

We now have to examine $\sum_{z' \in E} \Lambda_{zz'} H_{\mathcal{A}}(z')$ for an arbitrary partition \mathcal{A} of S . To this end, we use (2.7) and normalisation, followed by (2.45) and a change of summation involving (2.9) to calculate

$$\begin{aligned}
\sum_{z' \in E} \Lambda_{zz'} H_{\mathcal{A}}(z') &= \sum_{x, y \in \mathbb{X}} \lambda(z; y, x) [H_{\mathcal{A}}(z + \delta_x - \delta_y) - H_{\mathcal{A}}(z)] \\
&= \frac{(N - m)!}{N!} \sum_{\mathcal{B} \in \mathcal{O}_{\leq 2}(S)} r_{\mathcal{B}} \sum_{x, y \in \mathbb{X}} (R_{\mathcal{B}}(z))(x) z(y) [\bar{H}_{\mathcal{A}}(z + \delta_x - \delta_y) - \bar{H}_{\mathcal{A}}(z)] \\
&= \sum_{\mathcal{B} \in \mathcal{O}_{\leq 2}(S)} r_{\mathcal{B}} \left[\left(\frac{(N - (m - 1))!}{N!} \sum_{j \in M} \bar{H}_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{B}|_{A_j}} \right) - \frac{(N - m)!}{N!} m \bar{H}_{\mathcal{A}} \right] (z) \\
&= \sum_{\mathcal{B} \in \mathcal{O}_{\leq 2}(S)} r_{\mathcal{B}} \sum_{j \in M} (H_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{B}|_{A_j}} - H_{\mathcal{A}}) (z) \\
&= \sum_{j \in M} \sum_{\mathcal{J} \in \mathcal{O}_{\leq 2}(A_j)} \sum_{\substack{\mathcal{B} \in \mathcal{O}_{\leq 2}(S) \\ \mathcal{B}|_{A_j} = \mathcal{J}}} r_{\mathcal{B}} (H_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{J}} - H_{\mathcal{A}}) (z) \\
&= \sum_{j \in M} \sum_{\mathcal{J} \in \mathcal{O}_{\leq 2}(A_j)} r_{\mathcal{J}} (H_{\mathcal{A}_{M \setminus j}} \otimes R_{\mathcal{J}} - H_{\mathcal{A}}) (z),
\end{aligned}$$

which agrees with (2.44) and proves the claim. \square

We can now harvest some interesting consequences. First, Eq. (2.44) contains a meaningful expression for the derivative:

Corollary 2. *For $\mathcal{A} \in \mathbb{P}(S)$, $z \in E$, and the population process $\{Z_t\}_{t \geq 0}$, we have*

$$\frac{d}{dt} \mathbf{E}_\varphi [H_{\mathcal{A}}(Z_t) \mid Z_0 = z] \Big|_{t=0} = \sum_{j \in M} \sum_{\mathcal{J} \in \mathcal{O}_{\leq 2}(A_j)} r_{\mathcal{J}}^{A_j} \left(R_{\mathcal{J}}^{A_j} \otimes H_{\mathcal{A}_{M \setminus j}}^{A_{M \setminus j}} - H_{\mathcal{A}}^S \right)(z).$$

The right-hand side has a plausible explanation. Namely, when block A_j splits into \mathcal{J} , the other blocks in \mathcal{A} retain their current type distribution (namely, $H_{\mathcal{A}_{M \setminus j}}(\pi_{\mathcal{A}_{M \setminus j}} \cdot z)$). Independently of this, the parts of \mathcal{J} pick their types from *all* individuals (with replacement), including those individuals that already carry other parts of $\mathcal{A}_{M \setminus j}$, which is expressed by the tensor product with $R_{\mathcal{J}}(\pi_{A_j} \cdot z)$.

Next, via (2.39) together with the fact that $H_{\mathcal{A}}(z) = \mathbf{E}_\varphi [H_{\mathcal{A}}(Z_t) \mid Z_t = z]$, Eqns. (2.40) and (2.41) give rise to a system of differential equations for the expectations, namely:

Corollary 3. *For $\mathcal{A} \in \mathbb{P}(S)$ and the population process $\{Z_t\}_{t \geq 0}$, one has*

$$\frac{d}{dt} \mathbf{E}_\varphi [H_{\mathcal{A}}(Z_t)] = \sum_{B \in \mathbb{P}(S)} \Theta_{AB} \mathbf{E}_\varphi [H_B(Z_t)]. \quad (2.46)$$

This will be the basis for our concrete calculations in Section 2.8 and the numerical approach to estimate the recombination distribution in Chapter 3. However, first, we close this section with the conclusion that our model described by the dynamics (2.46) is *marginal consistent*.

Marginalisation consistency. This property was observed early in population genetics in [32] (see also the review [21]), but is not fulfilled in general for this kind of models, because of the influence of *selection*. However, in pure recombination models, the consistency usually holds and the marginalised dynamics is stated by the marginalised recombination rates as it was also shown recently in [3, Proposition 6] for the general recombination equation (in the infinite population size or deterministic limit) in continuous time. Our situation is tightly related to the latter and similar ideas are used to prove:

Proposition 6. *Let S be a finite set, $\emptyset \neq U \subset S$ and $\mathbf{E}_\varphi [H_{\mathcal{A}}^S(Z_t)]$ be the solution of (2.46) for $\mathcal{A} \in \mathbb{P}(S)$ with initial condition $H_{\mathcal{A}}^S(Z_0)$, $Z_0 \in E$. Then, for some $\underline{\mathcal{A}} \in \mathbb{P}(U)$ with $\underline{\mathcal{A}} = \mathcal{A}|_U$, the marginalised expected type distribution $\mathbf{E}_\varphi [H_{\underline{\mathcal{A}}}^U(\pi_U \cdot Z_t)]$ solves (2.46) for the subsystem given by U with operator as in (2.37).*

Proof. The verification is mainly based on two results stated in Section 2.6. First, we use the fact that the normalised sampling function (2.25) fulfils Proposition 4. Then, after using the linearity of π_U and the identity (2.46), we use the fact that the partitioning

process Σ_t with generator Θ is lumpable, see Proposition 5, to verify that

$$\begin{aligned}
\frac{d}{dt} \mathbf{E}_\varphi [H_A^U(\pi_U \cdot Z_t)] &= \pi_U \cdot \left(\frac{d}{dt} \mathbf{E}_\varphi [H_A^S(Z_t)] \right) = \pi_U \cdot \left(\sum_{\mathcal{B} \in \mathbb{P}(S)} \Theta_{\mathcal{A}\mathcal{B}} \mathbf{E}_\varphi [H_{\mathcal{B}}^S(Z_t)] \right) \\
&= \sum_{\mathcal{B} \in \mathbb{P}(S)} \Theta_{\mathcal{A}\mathcal{B}} \mathbf{E}_\varphi [\pi_U \cdot H_{\mathcal{B}}^S(Z_t)] = \sum_{\mathcal{B} \in \mathbb{P}(S)} \Theta_{\mathcal{A}\mathcal{B}} \mathbf{E}_\varphi [H_{\mathcal{B}|_U}^U(\pi_U \cdot Z_t)] \\
&= \sum_{\mathcal{B} \in \mathbb{P}(U)} \sum_{\substack{\mathcal{B} \in \mathbb{P}(S) \\ \mathcal{B}|_U = \mathcal{B}}} \Theta_{\mathcal{A}\mathcal{B}} \mathbf{E}_\varphi [H_{\mathcal{B}}^U(\pi_U \cdot Z_t)] \\
&= \sum_{\mathcal{B} \in \mathbb{P}(U)} \Theta_{\mathcal{A}\mathcal{B}} \mathbf{E}_\varphi [H_{\mathcal{B}}^U(\pi_U \cdot Z_t)]. \quad \square
\end{aligned}$$

Remark 9. *This is the key property that makes it even plausible to infer model parameters from some observation data as suggested in Chapter 3. Every used data (or sample) is always only a cutout of the entire system, since the data size from the entire genome would exceed any range and is usually even not available.*

2.8. Applications and examples

Let us now apply some of our results to the cases of $n = 2$ and $n = 3$ sites. Expectations will always be with respect to φ , so we will abbreviate \mathbf{E}_φ by \mathbf{E} throughout. We will assume that $Z_0 = z$, i.e., that the initial population is deterministic.

Two sites. For $U = S = \{1, 2\}$, with the abbreviation $r := r_{\{\{1\}, \{2\}\}}$, the ODE system of Corollary 3 reads

$$\begin{aligned}
\frac{d}{dt} \mathbf{E}[H_{\{\{1,2\}\}}(Z_t)] &= r \frac{N-1}{N} \mathbf{E}[(H_{\{\{1\}, \{2\}\}} - H_{\{\{1,2\}\}})(Z_t)] \quad (2.47) \\
\frac{d}{dt} \mathbf{E}[H_{\{\{1\}, \{2\}\}}(Z_t)] &= \frac{2}{N} \mathbf{E}[(H_{\{\{1,2\}\}} - H_{\{\{1\}, \{2\}\}})(Z_t)],
\end{aligned}$$

where we have dropped the upper index, which is always U . It follows that

$$\begin{aligned}
\frac{d}{dt} \mathbf{E}[(H_{\{\{1,2\}\}} - H_{\{\{1\}, \{2\}\}})(Z_t)] &= \\
&= - \left(\frac{2}{N} + r \frac{N-1}{N} \right) \mathbf{E}[(H_{\{\{1,2\}\}} - H_{\{\{1\}, \{2\}\}})(Z_t)]. \quad (2.48)
\end{aligned}$$

Since $L_{\{\{1,2\}\}} = \frac{N-1}{N} (H_{\{\{1,2\}\}} - H_{\{\{1\}, \{2\}\}})$, it follows that the expected two-point LDE also decays at rate $2/N + r(N-1)/N$. In the case of two alleles per site, an equivalent formula has appeared in [16, Ex. 1]. The corresponding result in the diffusion limit goes back to [74], see also [26, Chap. 8.2]. As noted there, it may seem surprising that the correlations also decay via resampling (even if $r = 0$); but recall that our Moran model with recombination is an absorbing Markov chain where a single type goes to fixation in

the long run, that is, Z_t will ultimately end up in a point measure.

The expected type distribution is now easily obtained from (2.47) and (2.48) via

$$\begin{aligned} \mathbf{E}[H_{\{\{1,2\}\}}(Z_t)] &= \mathbf{E}[H_{\{\{1,2\}\}}(Z_0)] - r \frac{N-1}{N} \int_0^t \mathbf{E}[(H_{\{\{1,2\}\}} - H_{\{\{1\},\{2\}\}})(Z_\tau)] d\tau \\ &= \frac{Z_0}{N} - \frac{r(N-1)}{r(N-1)+2} \left(1 - \exp\left(-\frac{r(N-1)+2}{N}t\right)\right) \mathbf{E}[(H_{\{\{1,2\}\}} - H_{\{\{1\},\{2\}\}})(Z_0)], \end{aligned}$$

where we have used that $\mathbf{E}[H_{\{\{1,2\}\}}(Z_0)] = Z_0/N$. The asymptotic behaviour is given by

$$\lim_{t \rightarrow \infty} \mathbf{E} \left[\frac{Z_t}{N} \right] = \frac{2}{2+r(N-1)} \frac{Z_0}{N} + \frac{r(N-1)}{2+r(N-1)} H_{\{\{1\},\{2\}\}}(Z_0). \quad (2.49)$$

Since Z_t will ultimately absorb in a point measure, we also know that

$$\lim_{t \rightarrow \infty} \mathbf{E} \left[\frac{Z_t}{N} \right] = \sum_{x \in \mathbb{X}} \mathbf{P}[Z_t \text{ absorbs in } x] \delta_x,$$

and thus $\mathbf{P}[Z_t \text{ absorbs in } x] = \lim_{t \rightarrow \infty} \mathbf{E}[Z_t/N](x)$ for all $x \in \mathbb{X}$. We can therefore read off the fixation probabilities from (2.49). With probability $\frac{2}{2+r(N-1)}$ (the relative intensity of resampling), the type that wins is drawn from the initial distribution. With probability $\frac{r(N-1)}{2+r(N-1)}$ (the relative intensity of recombination), it is drawn from the distribution that results when the leading and the trailing segments are sampled from the initial population without replacement.

Three sites. Now, we consider $U = S = \{1, 2, 3\}$, together with the abbreviations $r_1 := r_{\{\{1\},\{2,3\}\}}$, $r_2 := r_{\{\{1,2\},\{3\}\}}$ and $r_{\text{tot}} := r_{\text{tot}}(\mathbf{1}) = r_1 + r_2$. Let us order the partitions of $\mathbb{P}(U)$ as follows:

$$\{\{1, 2, 3\}\} \quad \{\{1\}, \{2, 3\}\} \quad \{\{1, 2\}, \{3\}\} \quad \{\{1, 3\}, \{2\}\} \quad \{\{1\}, \{2\}, \{3\}\}.$$

The generator of the partitioning process then reads

$$\Theta = \begin{pmatrix} -\frac{N-1}{N}r_{\text{tot}} & \frac{N-1}{N}r_1 & \frac{N-1}{N}r_2 & 0 & 0 \\ \frac{2}{N} - \frac{N-1}{N^2}r_2 & -\frac{2}{N} - \frac{(N-1)^2}{N^2}r_2 & \frac{N-1}{N^2}r_2 & \frac{N-1}{N^2}r_2 & \frac{(N-1)(N-2)}{N^2}r_2 \\ \frac{2}{N} - \frac{N-1}{N^2}r_1 & \frac{N-1}{N^2}r_1 & -\frac{2}{N} - \frac{(N-1)^2}{N^2}r_1 & \frac{N-1}{N^2}r_1 & \frac{(N-1)(N-2)}{N^2}r_1 \\ \frac{2}{N} - \frac{N-1}{N^2}r_{\text{tot}} & \frac{N-1}{N^2}r_{\text{tot}} & \frac{N-1}{N^2}r_{\text{tot}} & -\frac{2}{N} - \frac{(N-1)^2}{N^2}r_{\text{tot}} & \frac{(N-1)(N-2)}{N^2}r_{\text{tot}} \\ 0 & \frac{2}{N} & \frac{2}{N} & \frac{2}{N} & -\frac{6}{N} \end{pmatrix}. \quad (2.50)$$

Recall that, by Corollary 3 the expectation of $H(Z_t) := (H_A^U(Z_t))_{A \in \mathbb{P}(U)}$ fulfills $\frac{d}{dt} \mathbf{E}[H(Z_t)] = \Theta \mathbf{E}[H(Z_t)]$.

We now transform this system into a system in terms of correlation functions. Therefore, let $L(Z_t) = (L_A^U(Z_t))_{A \in \mathbb{P}(U)}$. From (2.30), we know that $L(Z_t) = TH(Z_t)$, where the

transformation matrix is given by

$$T = \frac{(N-1)(N-2)}{N^2} \begin{pmatrix} 1 & -1 & -1 & -1 & 2 \\ \frac{1}{N-2} & 1+\frac{1}{N-2} & \frac{-1}{N-2} & \frac{-1}{N-2} & -1 \\ \frac{1}{N-2} & -\frac{1}{N-2} & 1+\frac{1}{N-2} & -\frac{1}{N-2} & -1 \\ \frac{1}{N-2} & -\frac{1}{N-2} & -\frac{1}{N-2} & 1+\frac{1}{N-2} & -1 \\ \frac{1}{(N-1)(N-2)} & \frac{1}{N-2} & \frac{1}{N-2} & \frac{1}{N-2} & 1 \end{pmatrix}.$$

Consequently, $\frac{d}{dt} \mathbf{E}[L(Z_t)] = T\Theta T^{-1} \mathbf{E}[L(Z_t)]$, where

$$T\Theta T^{-1} = \begin{pmatrix} -\frac{6}{N} - \frac{(N-1)(N-2)}{N^2} r_{\text{tot}} & 0 & 0 & 0 & 0 \\ \frac{2}{N} - \frac{(N-1)}{N^2} r_{\text{tot}} & -\frac{2}{N} - \frac{N-1}{N} r_2 & 0 & 0 & 0 \\ \frac{2}{N} - \frac{(N-1)}{N^2} r_{\text{tot}} & 0 & -\frac{2}{N} - \frac{N-1}{N} r_1 & 0 & 0 \\ \frac{2}{N} - \frac{(N-1)}{N^2} r_{\text{tot}} & 0 & 0 & -\frac{2}{N} - \frac{N-1}{N} r_{\text{tot}} & 0 \\ -\frac{1}{N^2} r_{\text{tot}} & \frac{2}{N} - \frac{1}{N} r_2 & \frac{2}{N} - \frac{1}{N} r_1 & \frac{2}{N} - \frac{1}{N} r_{\text{tot}} & 0 \end{pmatrix}. \quad (2.51)$$

In contrast to (2.50), the matrix $T\Theta T^{-1}$ has a nice subtriangular structure, from which we can already read off that the *expected three-point LDE* $\mathbf{E}[L_{\{\{1,2,3\}\}}(Z_t)]$ (cf. (2.31)) decays exponentially according to

$$\frac{d}{dt} \mathbf{E}[L_{\{\{1,2,3\}\}}(Z_t)] = - \left(\frac{6N + (N-1)(N-2) r_{\text{tot}}}{N^2} \right) \mathbf{E}[L_{\{\{1,2,3\}\}}(Z_t)].$$

As in the case of two sites, the decay rate contains contributions from resampling as well as from recombination. To extract more information, we recast $T\Theta T^{-1}$ into the diagonal form $V^{-1}T\Theta T^{-1}V = D$, where the entries of the diagonal matrix D are those on the diagonal of $T\Theta T^{-1}$, i.e., its eigenvalues. Consequently, $\frac{d}{dt} V^{-1} \mathbf{E}[L(Z_t)] = DV^{-1} \mathbf{E}[L(Z_t)]$. With the help of the subtriangular structure of $T\Theta T^{-1}$, the matrix V^{-1} can be calculated explicitly. It is again subtriangular, but somewhat unwieldy. To streamline the results, we now turn to the diffusion limit, with generator Θ'' of Definition 2. Then T and T^{-1} converge to matrices T'' and $(T'')^{-1}$, respectively, with elements $T''_{\mathcal{A}\mathcal{B}} = \mu(\mathcal{B}, \mathcal{A}) \delta_{\mathcal{B} \preceq \mathcal{A}}$ and $(T'')^{-1}_{\mathcal{A}\mathcal{B}} = \delta_{\mathcal{B} \preceq \mathcal{A}}$, $\mathcal{A}, \mathcal{B} \in \mathbb{P}(U)$ (the latter is due to inversion from below). This yields

$$T''\Theta''(T'')^{-1} = \begin{pmatrix} -(6+\varrho_{\text{tot}}) & 0 & 0 & 0 & 0 \\ 2 & -(2+\varrho_2) & 0 & 0 & 0 \\ 2 & 0 & -(2+\varrho_1) & 0 & 0 \\ 2 & 0 & 0 & -(2+\varrho_{\text{tot}}) & 0 \\ 0 & 2 & 2 & 2 & 0 \end{pmatrix},$$

where $\varrho_i = \lim_{N \rightarrow \infty} N r_i$, $i \in \{1, 2, \text{tot}\}$. Note that the rescaling of time has already been absorbed in Θ'' . In place of V^{-1} , we now get

$$(V'')^{-1} = \begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 & 0 \\ \frac{1}{(2+\varrho_2)(4+\varrho_1)} & \frac{1}{2+\varrho_2} & 0 & 0 & 0 \\ \frac{2}{(2+\varrho_1)(4+\varrho_2)} & 0 & \frac{1}{2+\varrho_1} & 0 & 0 \\ \frac{1}{2(2+\varrho_{\text{tot}})} & 0 & 0 & \frac{1}{2+\varrho_{\text{tot}}} & 0 \\ \frac{4(\varrho_1\varrho_2+(2+\varrho_{\text{tot}})(6+\varrho_{\text{tot}}))}{(2+\varrho_1)(2+\varrho_2)(2+\varrho_{\text{tot}})(6+\varrho_{\text{tot}})} & \frac{2}{2+\varrho_2} & \frac{2}{2+\varrho_1} & \frac{2}{2+\varrho_{\text{tot}}} & 1 \end{pmatrix},$$

which diagonalises $T''\Theta''(T'')^{-1}$. This shows that, in contrast to $|U| = 2$, the linear combinations of $\mathbf{E}[L_{\mathcal{A}}(Z_t)]$'s, that decay exponentially, have coefficients depending on the recombination rates (with exception of $\mathbf{E}[L_{\{\{1,2,3\}\}}(Z_t)]$). As an example, $(4 + \varrho_1) \mathbf{E}[L_{\{\{1\}\{2,3\}\}}(Z_t)] + 2 \mathbf{E}[L_{\{\{1,2,3\}\}}(Z_t)]$ is one such combination and decays at rate $2 + \varrho_2$. Solution of the complete system is still possible due to the triangular structure; however, it is somewhat tedious since it involves the linear combination given in the last line of $(V'')^{-1}$. Further progress may be possible if alternative scalings are employed, such as the *loose linkage approach* suggested in [60].

2.9. Conclusion

Let us summarise our findings. We have described a marginal ancestral recombination process (ARP) and proved a duality result that relates the ARP with the Moran model forward in time, via so-called sampling functions. This was achieved by extending the recombinator formalism, which had previously proved useful in the context of deterministic recombination equations, to the stochastic setting. The ARP, together with the duality result, reveals the genealogical structure hidden in the work of [17], who approached the matter by functional-analytic means and forward in time. It also leads to an explicit and closed system of ordinary differential equations for the expected sampling functions, from which the expected linkage disequilibria of all orders can be calculated. It is quite remarkable that such a closed ODE system exists: after all, the sampling functions are nonlinear, and the attempt to write down the differential equation for the expectation of a nonlinear quantity usually results in a hierarchy of equations that does not close; see [6] for more on the moment closure problem in the case of recombination. We would like to emphasise that the favourable structure is due to the marginalisation, which gives efficient access to correlation functions, but not to variances, for example.

Unlike [17], we have not included mutation so far. However, since mutation acts independently of recombination, it should be straightforward to superimpose it on the population process as well as the partitioning process. It will be rewarding to study the interplay of mutation (which increases LDE) with recombination and resampling (which decrease LDE) within the framework established here.

Chapter 3.

Parameter estimation approach for the Moran model

3.1. Goal and outline

The purpose of this chapter is to propose and test a method to estimate the recombination probabilities as parameters of the Moran model with single crossover with respect to a given dataset. The tool of choice shall be a multiple shooting approach (see Appendix B) together with the duality result (Section 2.7) and its derived ODE (2.46).

In Section 3.2, first, a concrete *constrained discretised parameter least squares boundary value problem* is formulated in a nonlinear programming (NLP) framework. Then, it is restated for a special grid choice together with a rescaling of the recombination parameters. In Section 3.3, we consider the evaluation tasks of the NLP that may be requested by a solution algorithm (SQP)¹. We shed light on the structure of occurring matrices and propose an efficient storage and computing cascade on the NLP level. In Section 3.4, we give a detailed suggestion on how to implement the generator Θ (2.11) and the evaluation of the right hand side of (2.46), because this is the (by far) most frequently called program module during the numerical solution process. In Section 3.5, we see how (artificial) observation data for the least squares problem are generated for the numerical experiments in Section 3.6. The latter shall evidence the general applicability of the proposed approach in Section 3.2 for the purpose of estimating the (non-stationary) time course data determining recombination distribution.

For the programming we make use of a C++ library called Clean. Clean is an acronym for *A C++ Library for Efficient Algorithms in Numerics*, currently under development² in the Algorithmic Optimization group of Prof. M. C. Steinbach at Gottfried Wilhelm Leibniz Universität Hannover, and provides very efficient implementations of generic solvers for nonlinear optimisation problems.

¹A theoretical description of the sequential quadratic programming (SQP) approach and algorithm is given in Appendix A.2.

²It “[...] is not yet sufficiently mature but it is intended to become public domain as soon as it is considered” [82].

3.2. Moran model parameter estimation problem

Let $\mathbb{W} \subseteq \mathbb{X}$ be a subset of genetic types of cardinality $|\mathbb{W}|$. Then, for the ease of notation, we denote by

$$d_{\mathcal{A},\xi} : \mathbb{R}_{\geq 0} \rightarrow [0, 1], \quad d_{\mathcal{A},\xi}(t) := \mathbf{E}[H_{\mathcal{A}}(Z_t) \mid Z_0 = z](\xi) \in [0, 1], \text{ and} \quad (3.1a)$$

$$d : \mathbb{R}_{\geq 0} \rightarrow [0, 1]^{n_d}, \quad d(t) := \left((d_{\mathcal{A},\xi_1}(t))_{\mathcal{A} \in \mathbb{P}(S)}^T, \dots, (d_{\mathcal{A},\xi_{|\mathbb{W}|}}(t))_{\mathcal{A} \in \mathbb{P}(S)}^T \right)^T, \quad (3.1b)$$

the states of (2.46) with $n_d = b_n |\mathbb{W}|$. With this notations, (2.46) implies the linear system of ODEs $\dot{d}(t) = \left(I_{|\mathbb{W}|} \otimes \Theta(r) \right) d(t)$. Since recombination acts independent of the genetic types (recall the construction of the partitioning process in Section 2.4) this is a simple association of decoupled subsystems for each $\xi \in \mathbb{W}$ according to (3.1a).³ The data η of the (immediately formulated) least squares problem are assumed to be direct measurements of d . Further, let

$$h_j \in \mathbb{R}^{n_d} \quad (3.2a)$$

be the initial values in the discretisation nodes τ_j , $j = 1, \dots, n_\tau$ (cf. (B.5) and Figure B.1),

$$r = (r_{\mathcal{A}})_{\mathcal{A} \in \mathbb{O}_{\leq 2}(S)} \in \{p \in [0, 1]^n \mid \sum_{i=0}^{n-1} p_i = 1\} \quad (3.2b)$$

be the (only and global) vector of parameters, and

$$x := (h^T, r^T)^T := (h_1^T, \dots, h_{n_\tau}^T, r^T)^T \in \mathbb{R}^{n_\tau n_d + n}, \quad (3.2c)$$

the composition of the initial values and model parameters. So, besides the mandatory continuity constraints, the problem formulation only has to ensure that the probability distribution r stays feasible. This is incorporated as one (global) linear equation constraint (3.3c) together with (global) lower bounds for the parameters (3.3d) and completes the NLP

$$\min_x f(x) = \frac{1}{2} \sum_{i=1}^o \|d(t_i; h_{j(i)}, r) - \eta_i\|_{\Sigma_i^{-1}}^2, \quad (3.3a)$$

$$\text{s.t. } c_{E_{\text{co}}}(x) = \left(d(\tau_{j+1}; h_j, r) - h_{j+1} \right)_{j=1, \dots, n_\tau-1} = 0, \quad (3.3b)$$

$$c_{E_{\text{pd}}}(x) = \sum_{\mathcal{A} \in \mathbb{O}_{\leq 2}(S)} r_{\mathcal{A}} - 1 = 0, \quad (3.3c)$$

$$c_I(x) = r \geq 0, \quad (3.3d)$$

³The application of a second generator modelling mutation (and thus transitions between the $\xi \in \mathbb{W}$) would couple the subsystems and have a huge impact on the sparsity structures discussed in Section 3.3. But this is beyond the scope of this thesis.

where every (in-)equality has to be understood componentwise and $d(s; h_j, r)$ is the evaluation of the solution of the (local) *initial value problem* (IVP)

$$\dot{d}(t) = \left(I_{|\mathbb{W}|} \otimes \Theta(r) \right) d(t) \text{ with } d(\tau_j) = h_j \text{ for } j \leq s < j + 1, \quad (3.4)$$

for every $j = 1, \dots, n_\tau - 1$, each parametrised by (the same global) r . For the sake of structure and readability the indexset of equality constraints $E = E_{\text{co}} \dot{\cup} E_{\text{pd}}$ is separated in a subset E_{co} containing all nonlinear *continuity* constraints (due to the multiple shooting approach) and a singleton E_{pd} containing the sole linear constraint (due to the fact that r is a *probability distribution*).⁴

The NLP (3.3) is of the form (A.1) and not convex, since the equality constraints (3.3b) are not affine. Therefore local solutions are not necessarily global solutions (cf. [73] or [20]). The vector of inequality constraints c_I is not only linear in our model but even consists of simple lower bounds of the probabilities $r_{\mathcal{A}}$ only.⁵ When arranging the vector of dual multipliers accordingly as $\lambda = (\lambda_{\text{co}}, \lambda_{\text{pd}}, \lambda_I) \in \mathbb{R}^{(n_\tau - 1)n_d + 1 + n}$, the Lagrangian function (A.5) for this problem reads

$$\Lambda(x, \lambda) = f(x) - \lambda_{\text{co}}^T c_{E_{\text{co}}}(x) - \lambda_{\text{pd}} c_{E_{\text{pd}}}(x) - \lambda_I^T c_I(x). \quad (3.5)$$

Discretisation grid and scalable NLP. The NLP (3.3) makes no further assumptions on the covering discretisation grid $\{\tau_1, \dots, \tau_{n_\tau}\}$ than $[\tau_1, \tau_{n_\tau}] \supseteq [t_1, t_o]$. Taking $n_\tau < o$ discretisation nodes reduces the dimension of the NLP, but the evaluation of the objective f and its derivatives are more complicated, since we have to evaluate these terms for all initial values h . For the upcoming numerical experiments in Section 3.6, we choose the covering grid such that it coincides with the timepoints of observations, i.e. $n_\tau = o$ with $\tau_1 = t_1, \dots, \tau_{n_\tau} = t_o$. As a consequence, the objective is a *linear* least squares function and reads

$$f(x) = \frac{1}{2} \sum_{i=1}^o \|h_i - \eta_i\|_{\Sigma_i^{-1}}^2.$$

For computational reasons the NLP variables (3.2) should be of the same order of magnitude and not too close to 0 to prevent numerical difficulties (cf. [73, Chap. 2.2, 4.5]). In the current model formulation (3.3), especially the parameter vector $r = (r_{\mathbf{1}}, r_{\{\{1\}, \{2, \dots, n\}\}}, \dots, r_{\{\{1, \dots, n-1\}, \{n\}\}})$ may be poorly scaled. If we assume that

⁴By a slight abuse of notation the letter E has two canonical meanings. First, it is used for the state space of the Moran model with single crossover in Definition 1, and second, as the index set of equality constraints in the context of constrained optimisation, see the introduction of Section A.1. We do not use different symbols, since the context is always clear without ambiguity.

⁵We stick to the treatment as general inequality constraints here. The fact that they are of such a simple form is mainly of interest for the treatment of these constraints by a solution algorithm. We discuss this also later in the paragraph about the algorithmical evaluation process in Section 3.3.

$r_{\mathcal{A}} \leq 10^{-p}$ for all $\mathcal{A} \in \mathbb{O}_2(S)$ and some $p > \log_{10}(n-1)$, then

$$r_{\mathbf{1}} = 1 - \sum_{\mathcal{A} \in \mathbb{O}_2(S)} r_{\mathcal{A}} \geq 1 - (n-1)10^{-p} \gg r_{\mathcal{A}} \quad \forall \mathcal{A} \in \mathbb{O}_2(S).$$

To find a remedy, we use the abbreviation (recall the three sites example in Section 2.8, (2.14b), and (2.9))

$$r_{\text{tot}} := r_{\text{tot}}(\mathbf{1}) = \sum_{\mathcal{A} \in \mathbb{O}_2(S)} r_{\mathcal{A}}^S = 1 - r_{\mathbf{1}}^S = r_{\mathbf{0}}^{\{1,n\}},$$

apply the linear transformation

$$\begin{aligned} \phi : r &\mapsto \tilde{r} = (\tilde{r}_{\text{tot}}, \tilde{r}_{\{\{1\},\{2,\dots,n\}\}}, \dots, \tilde{r}_{\{\{1,\dots,n-1\},\{n\}\}})^T, \\ \phi(r) &:= \alpha(1 - r_{\mathbf{1}}, r_{\{\{1\},\{2,\dots,n\}\}}, \dots, r_{\{\{1,\dots,n-1\},\{n\}\}})^T, \end{aligned}$$

with $\alpha > 0$, and introduce the *scaled* NLP variable

$$\tilde{x} := (h^T, \tilde{r}^T)^T. \quad (3.6)$$

To obtain an equivalent reformulation of (3.3) with respect to (3.6) we make the following adjustment. Whenever (3.4) has to be evaluated, the transformation ϕ has to be inverted before updating the entries of the generator, because the latter depends on the original recombination distribution r . In addition, the last equality constraint (3.3c) now has to read $c_{E_{\text{co}}}(\tilde{x}) = \tilde{r}_{\text{tot}} - \sum_{\mathcal{A} \in \mathbb{O}_2(S)} \tilde{r}_{\mathcal{A}} = 0$; in order to preserve the condition that r is a probability distribution, we have to add an additional bound constraint $\tilde{r}_{\text{tot}} \in [0, \alpha]$. Then, for any $\alpha > 0$, the *scaled* formulation of the NLP with *adjusted discretisation grid* is⁶

$$\min_{\tilde{x}} f(\tilde{x}) = \frac{1}{2} \sum_{i=1}^o \|h_i - \eta_i\|_{\Sigma_i^{-1}}^2, \quad (3.7a)$$

$$\text{s.t. } c_{E_{\text{co}}}(\tilde{x}) = \left(d(\tau_{j+1}; h_j, \phi^{-1}(\tilde{r})) - h_{j+1} \right)_{j=1,\dots,n_{\tau}-1} = 0, \quad (3.7b)$$

$$c_{E_{\text{pd}}}(\tilde{x}) = \tilde{r}_{\text{tot}} - \sum_{\mathcal{A} \in \mathbb{O}_2(S)} \tilde{r}_{\mathcal{A}} = 0, \quad (3.7c)$$

$$c_I(\tilde{x}) = \begin{pmatrix} \tilde{r} \\ \alpha - \tilde{r}_{\text{tot}} \end{pmatrix} \geq 0. \quad (3.7d)$$

3.3. NLP evaluation and details of implementation

Now, we employ with the evaluation tasks a model implementation of the NLP (3.7) needs to perform at the request of a solution algorithm (we use a SQP algorithm), how this can be implemented, and discuss some numerical details. Therefore, we make use of the

⁶Note, that c_I now contains $n+1$ constraints and thus λ_I is of this increased dimension as well. But we refrain from highlighting this circumstance by altering the corresponding symbols.

notations introduced in Appendix A.1. And to distinguish the implementation from the mathematical formulation (NLP) we denote the first by NLP.

Evaluation tasks requested by a SQP algorithm. A generic solution algorithm for NLPs of the form (A.1) requests a model implementation to evaluate several parts of itself and, depending on the underlying concepts, derivatives of order 1 or higher. We omit a comprehensive discussion here. A short recapitulation of general nonlinear optimisation theory and a SQP algorithm is given in Appendix A. But we concretise our suggestion of NLP used for the numerical experiments in Section 3.6, which provides all necessary evaluations that may be requested by a SQP algorithm for a KKT point iterate $(\tilde{x}^{(k)}, \lambda^{(k)}) = (\tilde{x}^{(k)}, \lambda_{\text{co}}^{(k)}, \lambda_{\text{pd}}^{(k)}, \lambda_I^{(k)})$ as it is described in Section A.2. Those evaluations are:

$$f^{(k)} = f(\tilde{x}^{(k)}), \quad c_{E_{\text{co}}}^{(k)} = c_{E_{\text{co}}}(\tilde{x}^{(k)}), \quad c_{E_{\text{pd}}}^{(k)} = c_{E_{\text{pd}}}(\tilde{x}^{(k)}), \quad c_I^{(k)} = c_I(\tilde{x}^{(k)}), \quad (3.8a)$$

$$\nabla f^{(k)} = \nabla f(\tilde{x}^{(k)}), \quad C_{E_{\text{co}}}^{(k)} = \nabla c_{E_{\text{co}}}(\tilde{x}^{(k)}), \quad C_{E_{\text{pd}}}^{(k)} = \nabla c_{E_{\text{pd}}}(\tilde{x}^{(k)}), \quad C_I^{(k)} = \nabla c_I(\tilde{x}^{(k)}), \quad (3.8b)$$

and

$$\nabla_{\tilde{x}\tilde{x}}^2 \Lambda^{(k)} = \nabla_{\tilde{x}\tilde{x}}^2 \Lambda(\tilde{x}^{(k)}, \lambda^{(k)}). \quad (3.8c)$$

Obviously, (3.8a) covers the plain (partial) evaluations of (3.7); and (3.8b) are the Jacobians of the first. Since f is a quadratic form of the initial values h and, both, $c_{E_{\text{pd}}}$ and c_I are linear constraints of the parameters, their Jacobians pose no special computational challenge. For the evaluation of $c_{E_{\text{co}}}^{(k)}$ a numerical IVP solver has to be applied to evaluate $d(\tau_{j+1}; h_j^{(k)}, r^{(k)})$ first. How to do this in detail we discuss in the next paragraph, because this task is linked tightly to the evaluation of $C_{E_{\text{co}}}^{(k)}$ and $\nabla_{\tilde{x}\tilde{x}}^2 \Lambda^{(k)}$. In the following, the iteration index k is suppressed for the sake of readability. This is of no disadvantage, since all NLP evaluations only depend on the current iteration data.

Jacobian of continuity constraints. The Jacobian of the continuity constraint in a multiple shooting approach has a special structure independent of the underlying model. It is sparse, contains only two block diagonals (partial derivatives with respect to the initial values, two blocks for every subinterval j), and one block column (partial derivatives with respect to the parameters); see [19, Sec. 6.1] for a general discussion. An implementation should make use of the sparsity of the Jacobian to avoid the (constant) zero elements. Since we want to exploit this fact, we consider the internal structure of the blocks in more detail.

Figure 3.1 is an illustration of the Jacobian $C_{E_{\text{co}}}$ in our case. Every large block (belonging to a subinterval $[\tau_j, \tau_{j+1}]$) on the left diagonal is only a blockdiagonal submatrix itself, since the dynamic of the (local) IVP (3.4) is decoupled, because we do not model transitions

between the types ξ .⁷ Furthermore, $d(\tau_{j+1}; h_j, r)$ only depends on the data h_j , so it is possible to compute the blockrows $(C_{E_{\text{co}}})_j$ independent of each other. This enables to compute the parts sequentially (save memory) or parallel (save runtime). The proposed (overall) processing overview in Figure 3.4 provides for this fact.

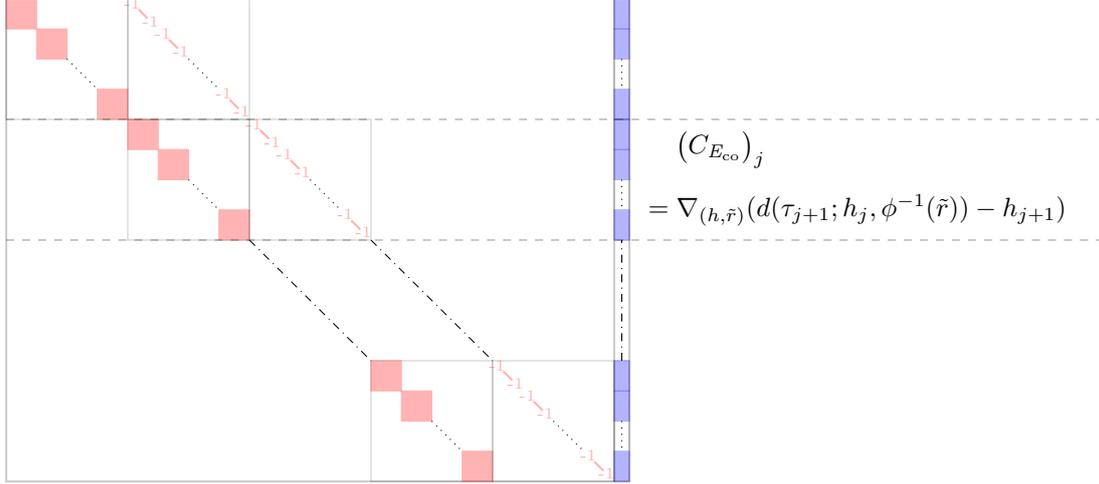


Figure 3.1.: Structure of the Jacobian of the continuity constraints: partial derivatives with respect to h (red) and with respect to \tilde{r} (blue). White areas contain zero elements.

The necessity of evaluating the Jacobian $C_{E_{\text{co}}}$ suggests to choose a code for *numerical solutions* of initial value problems (which has to be applied to evaluate $c_{E_{\text{co}}}$ anyways), that provides the possibility of computing the *sensitivity matrix* for the subinterval $[\tau_j, \tau_{j+1}]$,

$$G_j := \nabla_{(h_j, \tilde{r})} d(\tau_{j+1}; h_j, \phi^{-1}(\tilde{r})) = \begin{bmatrix} G_{j, \xi_1}^h & & G_{j, \xi_1}^{\tilde{r}} \\ & \ddots & \vdots \\ & & G_{j, \xi_{|W|}}^h & G_{j, \xi_{|W|}}^{\tilde{r}} \end{bmatrix}, \quad (3.9a)$$

with non-zero blocks

$$G_{j, \xi}^h := \nabla_{h_j, \xi} \left(d(\tau_{j+1}; h_j, \phi^{-1}(\tilde{r})) \right)_{\xi} \quad \text{and} \quad (3.9b)$$

$$G_{j, \xi}^{\tilde{r}} := \nabla_{\tilde{r}} \left(d(\tau_{j+1}; h_j, \phi^{-1}(\tilde{r})) \right)_{\xi} \quad (3.9c)$$

for all ξ , if requested. The code METAN ([10]), that has been implemented for PARFIT ([18]) satisfies this requirement and, therefore, it is used in Section 3.6. It applies *automatic differentiation* for the computation of the sensitivity matrix and is based on a *Richardson extrapolation* approach applying a *modified semi-implicit midpoint-rule*.⁸

⁷If a generator modelling mutation would be applied to the dynamics additionally, this block would be filled with non-zero values as well as the subsystems in (3.4) would not be decoupled anymore.

⁸An alternative from the same class, and also part of PARFIT, is DIFSYS, which relies on a *modified explicit midpoint-rule*. It is also called *Bulirsch-Stoer algorithm* or *Gragg-Bulirsch-Stoer algorithm*, because it

Hessian of the Lagrangian function. Obviously, the direct evaluation of the Hessian of the Lagrangian function is not possible, since we do not have access to an analytical formulation of the latter; we are already dependent on numerical evaluations of (3.7b) and its Jacobian. That's why we need to use a practical approximation of $\nabla_{\tilde{x}\tilde{x}}^2 \Lambda$ (positiv semidefinit if possible). For the purpose of the experiments in this thesis we use the approach⁹

$$\begin{aligned} \nabla_{\tilde{x}\tilde{x}}^2 \Lambda &= \nabla_{\tilde{x}\tilde{x}}^2 f(\tilde{x}) - \nabla_{\tilde{x}\tilde{x}}^2 \lambda_{\text{co}}^T c_{E_{\text{co}}}(\tilde{x}) - \nabla_{\tilde{x}\tilde{x}}^2 \lambda_{\text{pd}} c_{E_{\text{pd}}}(\tilde{x}) - \nabla_{\tilde{x}\tilde{x}}^2 \lambda_I^T c_I(\tilde{x}) \\ &= \begin{bmatrix} \Sigma^{-1} & 0 \\ 0 & 0 \end{bmatrix} - \sum_{(j,\xi,\mathcal{A})} \lambda_{\text{co};(j,\xi,\mathcal{A})} \left[\nabla_{\tilde{x}\tilde{x}}^2 c_{E_{\text{co}}}(\tilde{x}) \right]_{(j,\xi,\mathcal{A})} \\ &\approx \begin{bmatrix} \Sigma^{-1} & 0 \\ 0 & 0 \end{bmatrix} - \sum_{(j,\xi,\mathcal{A})} \lambda_{\text{co};(j,\xi,\mathcal{A})} \left[C_{E_{\text{co}}}(\tilde{x})^T C_{E_{\text{co}}}(\tilde{x}) \right]_{(j,\xi,\mathcal{A})}. \end{aligned} \quad (3.10)$$

The symmetric and sparse structure of this approximation is illustrated in Figure 3.2. The blocks are coloured with respect to the different type of summands in (3.10); the distinction is based on the (products) of partial derivatives that make non-zero contributions to a corresponding block.

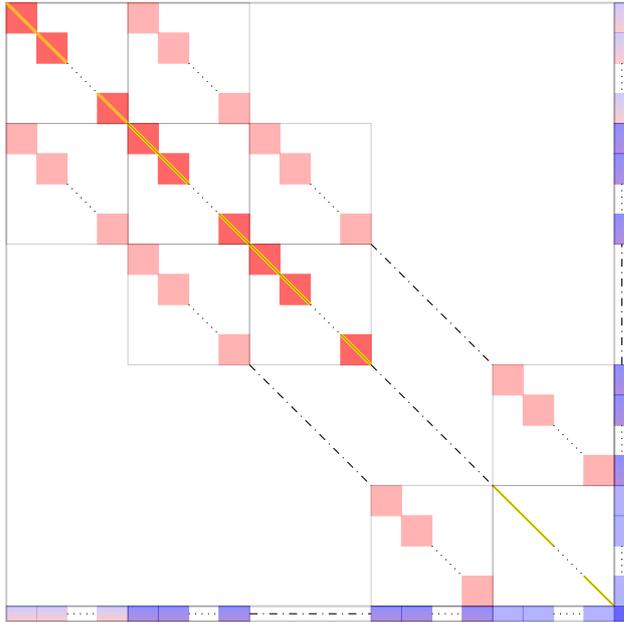


Figure 3.2: Structure of the $\nabla_{\tilde{x}\tilde{x}}^2 \Lambda$ approximation. Blocks are coloured by their dependence on: partial derivatives of $c_{E_{\text{co}}}$ with respect to h (light red) and with respect to \tilde{r} (light blue); rank-1 (red, blue) and mixed (shaded purple) products of the previous; $\nabla_{hh}^2 f = \Sigma^{-1}$ (yellow); additional diagonal part dependent on λ_{co} only (solid black). The remaining elements are constant zero (white).

Remark 10 (Approximation approach for Hessian of the Lagrangian function). *The approach in (3.10) is used in unconstrained least squares approaches (Gauß-Newton), cf. [73, Chapter 10]. Of course, $c_{E_{\text{co}}}$ is not a least squares function, but only componentwise*

first makes use of a result by William B. Gragg [51] to avoid asymptotical oscillation of the discretisation error function. METAN, as a later development, makes us of this result as well.

⁹Note, that the multiindex (j, ξ, \mathcal{A}) at the Hessian of the continuity constraints (and its approximation) identifies a summand (and hence a matrix) and is not meant as the selection of a matrix element.

almost zero in a KKT point (\tilde{x}^*, λ^*) , so this approach may be somewhat naive and other (more sophisticated) approximations or updates (like SR1, cf. [73, Chapter 6]) should be tested. But, it turns out to be a somewhat useful choice here, see the results in Section 3.6.

Datastructure of the derivatives and the algorithmical evaluation process. As it was already emphasised in the preceding paragraphs, the Jacobians of the constraints and the Hessian of the Lagrangian function are sparse. The latter is symmetric as well, which allows to reduce the amount of necessary memory by almost 50% additionally. Therefore, it is worthwhile to use an appropriate datastructure for these objects and we choose a (symmetric) triplet sparse format in our implementation of $C_{E_{co}}(\nabla_{\tilde{x}\tilde{x}}^2 A)$.

Remark 11. Note that this efficient memory usage makes sense, because the dimension $n_\tau b_n |\mathbb{W}| + n$ of the NLP variable and the number of constraints $(n_\tau - 1) b_n |\mathbb{W}| + n + 2$ grow superexponentially with the number of considered sites $n = |S|$ (due to the Bell number b_n , cf. Table 2.1).

Now, let us briefly discuss how to aggregate the evaluations (3.8) in an algorithm. Our NLP has to evaluate the requested parts of (3.8) according to a list of *tasks* and a *current KKT iterate* (\tilde{x}, λ) , that are provided by a SQP algorithm. Therefore, by (3.10), we note that $\nabla_{\tilde{x}\tilde{x}}^2 A$ only depends on the constant part Σ^{-1} and the current evaluation of $C_{E_{co}}$. The latter can be computed independently for every subinterval. Thus, after having calculated G_j , the update of $\nabla_{\tilde{x}\tilde{x}}^2 A$ (with respect to subinterval j) takes place as illustrated in Figure 3.3.

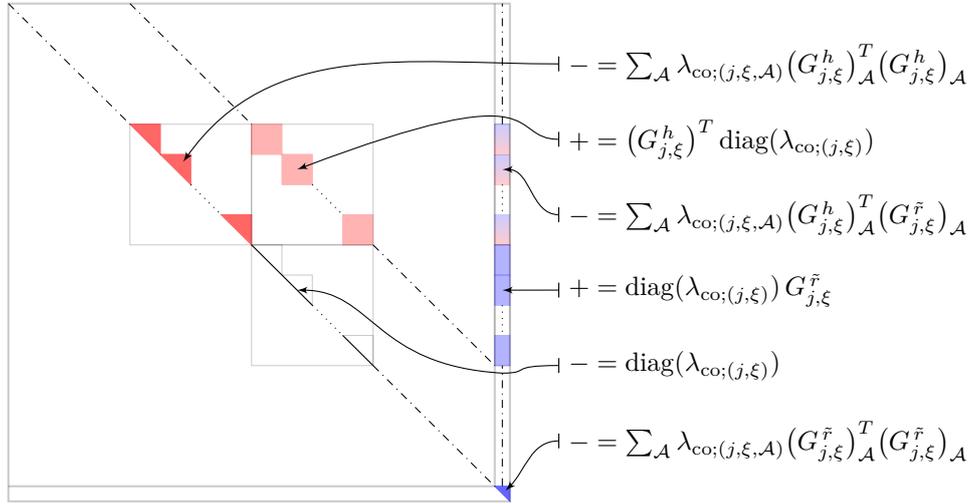


Figure 3.3.: Increment of (the approximation (3.10) of) $\nabla_{\tilde{x}\tilde{x}}^2 A$ by evaluations depending on the subinterval $[\tau_j, \tau_{j+1}]$: $\text{diag}(\lambda_{\text{co};(j,\xi)})$ means the diagonal matrix of size $b_n \times b_n$ with elements $(\lambda_{\text{co};(j,\xi,\mathcal{A})})_{\mathcal{A}}$; $(G_{j,\xi}^{\bullet})_{\mathcal{A}}$ is a row vector, so $(G_{j,\xi}^{\bullet})_{\mathcal{A}}^T (G_{j,\xi}^{\bullet})_{\mathcal{A}}$ is a rank-1 matrix; the constant part Σ^{-1} (cf. Figure 3.2) can be added in a preprocessing already.

Summarised, NLP has to run through the cascade in Algorithm 1 (a slightly more detailed illustration thereof is given in Figure 3.4). Recall that (3.7d) are simple bound constraints,

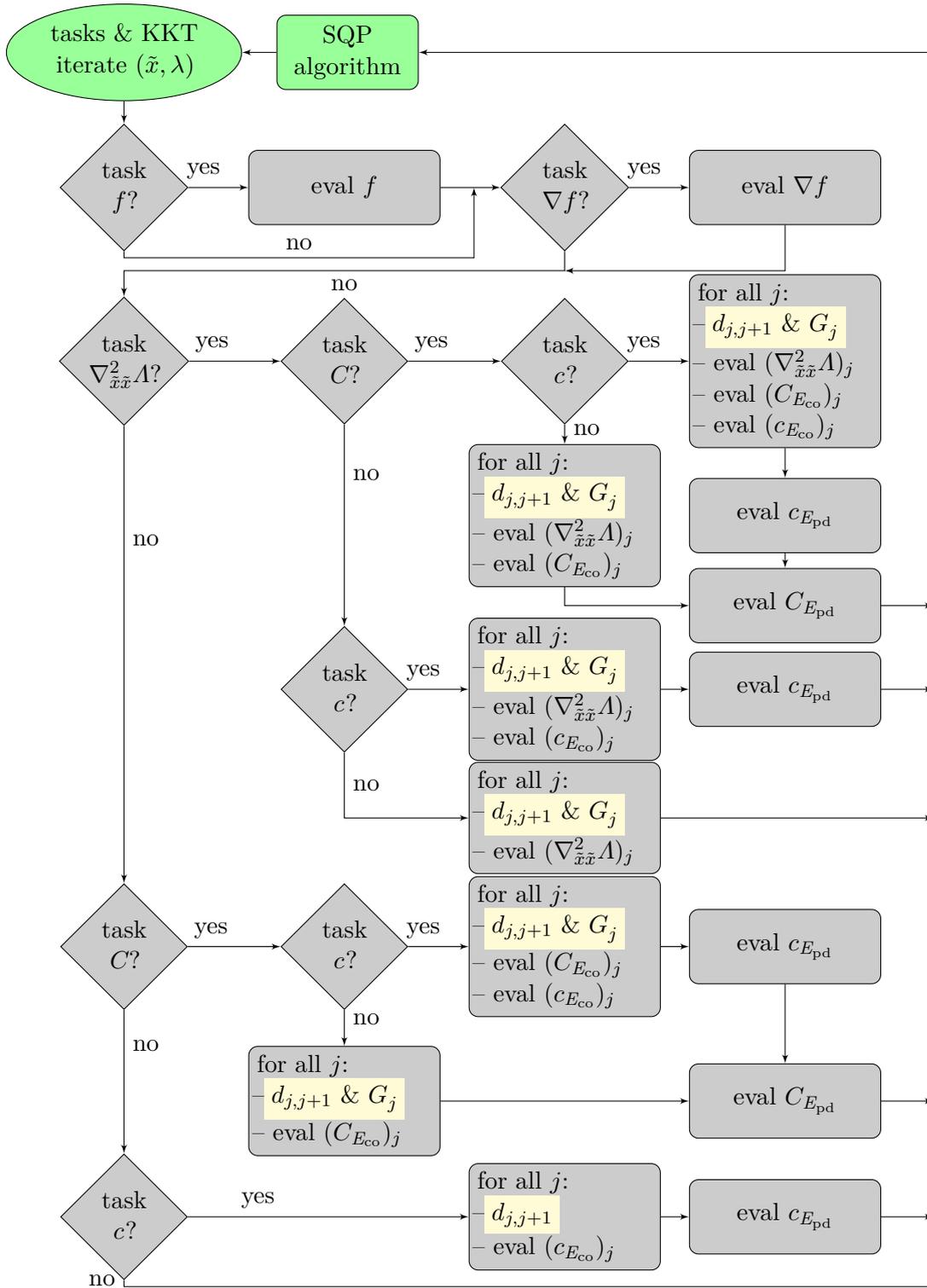


Figure 3.4.: Flowchart of NLP related evaluations (3.8) (grey) depending on a list of SQP tasks and current KKT iterates (green); required IVP solution evaluation $d_{j,j+1} := d(\tau_{j+1}; h_j, \phi^{-1}(\tilde{r}))$ (3.4) and corresponding sensitivity matrix (3.9) (yellow).

Algorithm 1: NLP evaluation cascade

Input : Current KKT iterate (\tilde{x}, λ) and SQP tasks.

- 1 Evaluate f or ∇f , if requested.
 - 2 Check whether $\nabla_{xx}^2 \Lambda$ or $C_{E_{co}}$ is requested.
 - 3 If yes, evaluate $d(\tau_{j+1}; h_j, \phi^{-1}(\tilde{r}))$ together with $G_j \forall j = 1, \dots, n_\tau - 1$ and update $\nabla_{\tilde{x}\tilde{x}}^2 \Lambda$, $C_{E_{co}}$ or $c_{E_{co}}$, if requested.
 - 4 If not, check whether $c_{E_{co}}$ is requested.
 - 5 If yes, evaluate $d(\tau_{j+1}; h_j, \phi^{-1}(\tilde{r}))$ without $G_j \forall j = 1, \dots, n_\tau - 1$ and update $c_{E_{co}}$.
 - 6 Evaluate $c_{E_{pd}}$, if requested.
 - 7 **return** results to SQP algorithm.
-

which are handled by `Clean::SQP` automatically (via a specific interface). Since they also do not occur in (3.10), we do not have to treat their evaluation in NLP.

As it should be clear now, for an implementation of the NLP (3.3) (or (3.7)) the evaluation of (3.4) is one of the crucial and the by far most frequent task. Everytime a NLP solver (SQP) requests to evaluate its solution (or their sensitivity matrix) for any $s \in [\tau_j, \tau_{j+1}]$, $j = 1, \dots, n_\tau - 1$, the corresponding numerical integration method has to evaluate the ODE several times for some fixed iterate r , which thus belongs to the core building blocks of NLP. Therefore, it is worth to put effort into an implementation, which makes this task as *efficient* as possible on the one hand, and pays regard on the *numerics* on the other hand. The proposed datastructure and algorithms to attain these goals are described in the following Section 3.4 and, since they are based on the results of Section 2.4, especially the datastructure deviates substantially from the dense representation referred to in [17] and is not only a simple transfer of the latter into an ordinary sparse version.

3.4. Matrix representation of the generator and evaluation of the ODE system

As already emphasised in Section 2.4, the ratio of non-zero transitions of the partitioning process is decreasing exponentially with the increase of the number of sites n , see Table 2.1. Therefore, in principle, a sparse matrix format omitting the zero valued elements is recommended. The special one proposed here is derived from a triplet sparse format, which only holds a list of triplets (row index, column index, non-zero value).

Apart from that, the non-zero entries of Θ share some additional structure that is slightly hidden in the form of (2.11) and (2.14), and therefore we elaborate on this first. Then, we formulate the specialised datastructure and associated algorithms that take advantage of this elaborated rearrangement.

Rearrangement of the generator. First, interpret Θ as a linear combination in the order of magnitudes $0, -1, -2$ of the constant population size N and split the entries

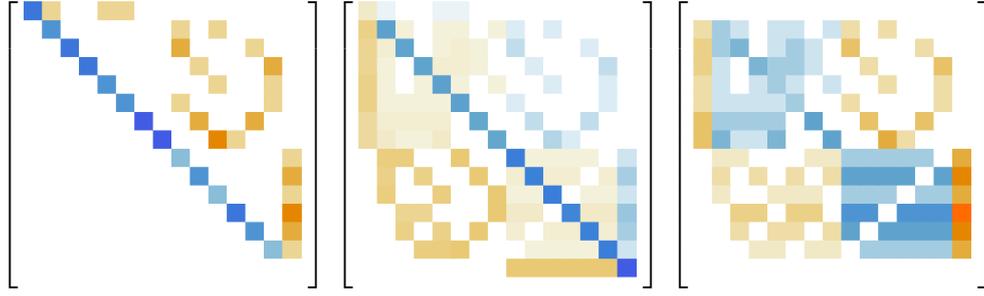


Figure 3.5.: From left to right: heatmap of the coefficient matrices $\Theta^{[0]}(r)$, $\Theta^{[1]}(r)$, $\Theta^{[2]}(r)$ for $r = (0.997500, 0.000833, 0.000833, 0.000833)$ bringing out their sparsity structures. White areas represent constant zero elements.

correspondingly. It is

$$\Theta(r; N) = \Theta^{[0]}(r) + N^{-1}\Theta^{[1]}(r) + N^{-2}\Theta^{[2]}(r) \quad (3.11)$$

and an example of the sparsity structures of the coefficient matrices for $n = 3$ is given in Figure 3.5. The notation shall emphasise the inherent dependance of the generator $\Theta(r; N)$ on the population size compared to the *coefficient matrices* $\Theta^{[i]}(r)$, $i = 0, 1, 2$. For each $\mathcal{A} \in \mathbb{P}(S)$, $|\mathcal{A}| = m$, the occurrence of transitions in (2.11) (and their corresponding rate) are subdivided with respect to $|\mathcal{B}|$ into

$$\left. \begin{array}{l} \mathcal{B}|_{A_j} = \mathcal{J}, \mathcal{B}|_{A_M \setminus j} = \mathcal{A}_{M \setminus j} \\ \text{for some } j \in M, \mathcal{J} \in \mathbb{O}_2(A_j) \end{array} \right\} \Rightarrow \begin{cases} r_{\mathcal{J}}^{A_j} \frac{1}{N^2}, & |\mathcal{B}| = m - 1, \\ r_{\mathcal{J}}^{A_j} \left(\frac{1}{N} - \frac{m-1}{N^2} \right), & |\mathcal{B}| = m, \\ r_{\mathcal{J}}^{A_j} \left(1 - \frac{2m-1}{N} + \frac{m(m-1)}{N^2} \right), & |\mathcal{B}| = m + 1. \end{cases} \quad (3.12)$$

In doing so, each subcase occurs $(n - m)(m - 1)(m - 2)$, $(n - m)(2m - 2) + 1$, and $n - m$ times, respectively. The coalescence case (2.12) splits into

$$\left. \begin{array}{l} \mathcal{B} = \mathcal{A}_{M \setminus \{j,k\}} \cup \mathcal{A}_{\{j,k\}} \\ \text{for some } j \neq k \in M \end{array} \right\} \Rightarrow \begin{cases} \frac{2}{N}, & |A_{\{j,k\}}| = 2, \\ \frac{1}{N}(r_1^{A_j} + r_1^{A_k}) - \frac{1}{N^2}(-2 + r_1^{A_j} + r_1^{A_k}), & \text{otherwise.} \end{cases} \quad (3.13)$$

With $\sigma(\mathcal{A}) := |\{A \in \mathcal{A} \mid |A| = 1\}|$ counting the number of singletons of a partition \mathcal{A} , the two subcases in (3.13) occur $\binom{\sigma(\mathcal{A})}{2}$ and $\binom{m}{2} - \binom{\sigma(\mathcal{A})}{2}$ times each. Finally, and for the sake of completeness, (2.14a) is rearranged as

$$\Theta_{\mathcal{A}\mathcal{A}}(r; N) = - \left[r_{\text{tot}}(\mathcal{A}) + \frac{m(m-1) - m r_{\text{tot}}(\mathcal{A})}{N} + \frac{m-1}{N^2} r_{\text{tot}}(\mathcal{A}) \right]. \quad (3.14)$$

Remark 12 (Numerical aspect). *The form (3.11) provides already a numerical advantage. For every k , the coefficient matrix $\Theta^{[k]}(r)$ has non-zero elements that are a product (sum)*

of an integer $< n^2$ with (and) a real value in $[0, 2]$. Therefore, they are of similar magnitude and thus the magnitude of each summand in (3.11) is determined by N^{-k} . Now, when applying $\Theta(r; N)$ as a linear map, it is possible to add the components in ascending order of magnitude to minimise rounding errors.

Datastructure and associated algorithms. The subdivided cases in the previous paragraph provide a possible saving in updating the matrix entries *after* an update of r , since the coefficient matrices $\Theta^{[0]}(r)$, $\Theta^{[1]}(r)$, and $\Theta^{[2]}(r)$ share common marginal recombination probabilities $r_{\mathcal{J}}^{A_j}$ and $r_{\text{tot}}(\mathcal{A})$, respectively. They are *weighted sums* of the current values of r . Hence, it appears reasonable to represent the *structure* of each $\Theta^{[k]}(r)$ by an array of n_k quadrupels

$$\Theta_{i_k}^{[k]} = (\mathcal{A}, \mathcal{B}, c, i_W), \quad (3.15)$$

which contains the two states of the transition (row and column index), an integer c capturing the constant part of each particular numerator, and an index i_W linking to an array containing the corresponding (shared) weighted sums of r . Consequently, this structure is *independent* of the values of r and does not contain any transition rates yet. Those are made persistent in an additional (*data-*) array $\theta^{[k]} \in \mathbb{R}^{n_k}$ of the same size, such that, by a slight abuse of notation, the separation $\Theta^{[k]}(r) \hat{=} (\Theta^{[k]}, \theta^{[k]})$ in structural and data elements is possible for $k = 0, 1, 2$. Taking the numbers of each subcase in (3.12) to (3.14) into account, the sizes of these arrays are

$$n_0 = \sum_{m=1}^n \binom{n}{m} (n-m) + (b_n - 1), \quad (3.16a)$$

$$n_1 = \sum_{m=1}^n \binom{n}{m} \left[(n-m)(2m-1) + \binom{m}{2} \right] + b_n, \quad \text{and} \quad (3.16b)$$

$$n_2 = \sum_{m=3}^n \binom{n}{m} \left[\binom{m}{2} (2(n-m) + 1) + (n-m) \right] + (b_n - 2) + \binom{n}{2} (3n-5) - \sum_{\mathcal{A} \in \mathbb{P}(S)} \binom{\sigma(\mathcal{A})}{2}. \quad (3.16c)$$

The approach of separating the structure from values is also applied to the weighted sums of r , which are necessary to compute $\theta^{[k]}$. The weights are constant and depend only on the transition $\mathcal{A} \rightarrow \mathcal{B}$, so they are captured in n_W *weight arrays* $w_{i_W} \in \mathbb{Z}^n$ with

$$n_W = \sum_{m=1}^n \binom{n}{m} \left[(n-m) \left(2 \binom{m}{2} + 1 \right) + \binom{m}{2} \right] + b_n - \sum_{\mathcal{A} \in \mathbb{P}(S)} \binom{\sigma(\mathcal{A})}{2}, \quad (3.16d)$$

which are combined to the *structure* matrix

$$W = (w_{i_W}^T) \in \mathbb{Z}^{n_W \times n}. \quad (3.17)$$

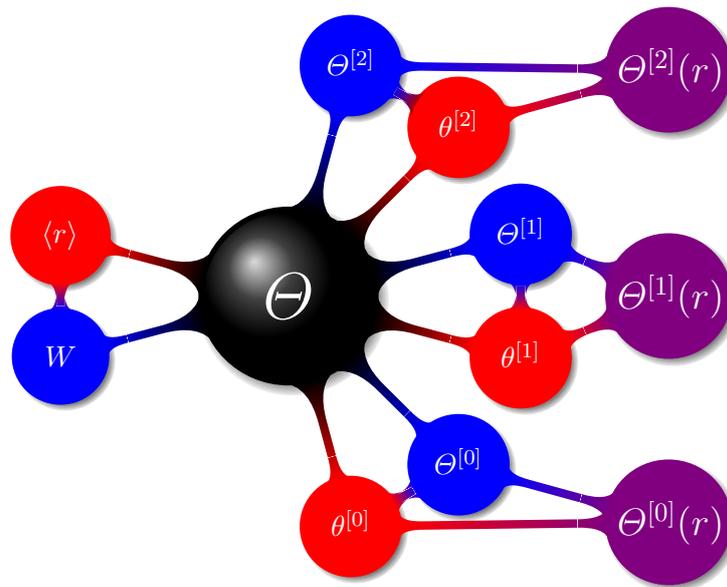


Figure 3.6.: Illustration of the concept of implementation for Θ . Structural elements are indicated in blue, data elements in red and coefficient matrices in purple. When Θ has to be updated, first, $\langle r \rangle$ is calculated with respect to W , followed by $\theta^{[k]}$ with respect to $\Theta^{[k]}$. Afterwards, $\Theta^{[k]}(r)$ can be used in the linear map provided by (3.11).

Concluding, the corresponding *data* array containing the current evaluation of all necessary cases is denoted by $\langle r \rangle = Wr \in \mathbb{R}^{nw}$; and Wr is meant in the sense of an ordinary matrix-vector-product. In Figure 3.6, we give an overview of the proposed datastructure. Structural elements are coloured blue, data elements are coloured red and functional elements (as applications of the coefficient matrices $\Theta^{[i]}(r)$) are coloured purple.

Summarising, a model implementation using this representation of the generator has the possibility to use the following features:

- Updating the weighted sums of recombination parameters $\langle r \rangle$ and the values of $\theta^{[0]}$, $\theta^{[1]}$, and $\theta^{[2]}$ separately, but not in every evaluation of the ODE. This is only necessary when a new iterate r is produced and during the process of calculating the sensitivity matrices G_j (3.9) for every j .
- Application of (3.11) as a linear map *sequentially* in ascending order of the magnitude of N with the possibility to *truncate terms* of some order if desired, see Algorithm 4. Every truncation of magnitude $k = 1, 2$ increases the sparsity of $\Theta(r; N)$ and directly reduces the amount of arithmetic operations by $\sum_{l=k}^2 n_l$.¹⁰

Furthermore, let us quantify the improvement of the proposed datastructure based on the necessary amount of memory compared to the dense approach of [17], which relies on the decomposition (2.18) of Θ with respect to all splitpositions between two consecutive

¹⁰The case $k = 0$ would correspond to $\Theta(r; N) \equiv 0$.

sites. Their version needs to hold n dense (constant, if N is constant) matrices with double entries, so the amount of memory sums up to $8nb_n^2$ bytes. Our sparse approach needs $4(4(n_0 + n_1 + n_2) + nn_W)$ bytes for the structural information ($\Theta^{[k]}$ for $k = 0, 1, 2$ and W) plus $8(n_0 + n_1 + n_2 + n_W)$ bytes for the double values ($\theta^{[k]}$ for $k = 0, 1, 2$ and $\langle r \rangle$). Figure 3.7 illustrates the increasing gap between these two alternatives for $n > 4$; up to $n = 4$ the total amount of memory is negligible in any case.

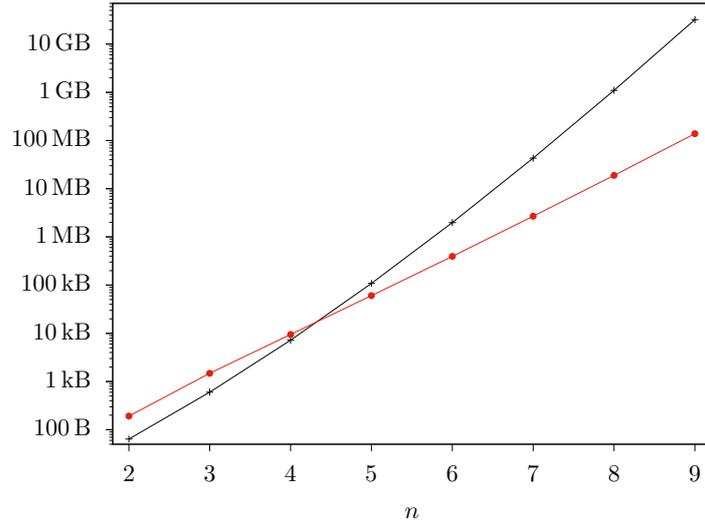


Figure 3.7.: Necessary amount of memory for the dense [17] (black crosses) and our sparse (red bullets) datastructure to persist Θ .

Last, we give the algorithmic construction (Algorithm 2) and update (Algorithm 3) of Θ as well as its induced linear map (Algorithm 4) based on the proposed structure. We provide the formulated features together with some closing remarks.

Remark 13. *One of the benefits of using a sorted list of partitions, when constructing Θ in Algorithm 2, is that the inner loop in line 7 of Algorithm 2 only runs through the (smaller) subsets $\mathbb{P}_\ell(S)$, $|\mathbb{P}_\ell(S)| = \binom{\ell}{2}$ for $\ell = |\mathcal{A}| - 1, |\mathcal{A}|, |\mathcal{A}| + 1$, instead of the entire set $\mathbb{P}(S)$. This is because the possible states \mathcal{B} , that can be reached from \mathcal{A} , are more aggregated. Moreover, note that the constant part of the coefficients can not always be stored directly in the structure $\Theta^{[k]}$ (for example in line 12 of Algorithm 2), but the correct form is recovered with the update process in line 16 of Algorithm 3.*

Remark 14. *Furthermore, note that we really have to rely on the automatic differentiation when computing the sensitivity matrices in (3.9), although the right hand side of (3.4) is linear in r , and one could be tempted to make use of this rearrangement there. But the semigroup $\exp(\Theta(r; N)t)$ does not commute with the (constant) partial derivatives $\frac{d}{dr_{\mathcal{A}}} \Theta(r; N)$.*

Algorithm 2: Constructor of the generator's structure (3.15) and (3.17)

Input : The number of sites $n = |S|$ and $\mathbb{P}(S)$ as a list in ascending order of $m = |\mathcal{A}|$.

```

1  Set number of entries  $n_0, n_1, n_2, n_W$  according to (3.16).
2  Initialise indices  $i_0, i_1, i_2, i_W$  and set  $m = 1$ .
3  forall  $\mathcal{A} \in \mathbb{P}(S)$  do
4      if  $|\mathcal{A}| > m$  then
5          |  $m \leftarrow |\mathcal{A}|$ .
6      else
7          forall  $\mathcal{B} \in \bigcup_{\ell=m-1}^{m+1} \mathbb{P}_\ell(S)$  do
8              if  $\mathcal{A} = \mathcal{B}$  then
9                  // Diagonal element (3.14)
10                  $W_{i_W} \leftarrow w^T$  with  $w^T$  according to (2.14b).
11                  $\Theta_{i_2}^{[2]}, \Theta_{i_0}^{[0]}$  w.r.t. (3.14) and with index  $i_W$ .
12                  $i_k \leftarrow i_k + 1$  if  $\Theta_{i_k}^{[k]}$  was set in line 10.
13                  $\Theta_{i_1}^{[1]} \leftarrow (\mathcal{A}, \mathcal{A}, m, i_W)$ ,  $i_1 \leftarrow i_1 + 1$ .
14                  $i_W \leftarrow i_W + 1$ .
15             else
16                  $\delta_{\text{coal}} \leftarrow \text{false}$ .
17                 forall  $j = 1, \dots, m$  do
18                     if  $\mathcal{B}|_{S \setminus A_j} = \mathcal{A}|_{M \setminus j}$  then
19                         if  $!\delta_{\text{coal}} \ \& \ \mathcal{B} \succ \mathcal{A}$  then
20                             // Coalescence element (3.13)
21                             Identify  $B = A_{\{j,k\}}$ .
22                             if  $|B| = 2$  then
23                                 |  $\Theta_{i_1}^{[1]} \leftarrow (\mathcal{A}, \mathcal{B}, 2, \emptyset)$ ,  $i_1 \leftarrow i_1 + 1$ .
24                             else
25                                 | Set  $A_k = B \setminus A_j$ .
26                                 |  $W_{i_W} \leftarrow (\delta_{s \dagger A_j} + \delta_{s \dagger A_k})_{s=0, \dots, n-1}$ .
27                                 |  $\Theta_{i_1}^{[1]} \leftarrow (\mathcal{A}, \mathcal{B}, +1, i_W)$ ,  $i_1 \leftarrow i_1 + 1$ .
28                                 |  $\Theta_{i_2}^{[2]} \leftarrow (\mathcal{A}, \mathcal{B}, -1, i_W)$ ,  $i_2 \leftarrow i_2 + 1$ .
29                                 |  $i_W \leftarrow i_W + 1$ .
30                              $\delta_{\text{coal}} \leftarrow \text{true}$ .
31                         if  $|A_j| > 1$  then
32                             forall  $\mathcal{J} \in \mathbb{O}_2(A_j)$  do
33                                 if  $\mathcal{B}|_{A_j} = \mathcal{J}$  then
34                                     // Split (or mixed) element (3.12)
35                                      $W_{i_W} \leftarrow (\delta_{\max \mathcal{J}_1 \leq s < \min \mathcal{J}_2})_{s=0, \dots, n-1}$ .
36                                      $\Theta_{i_2}^{[2]}, \Theta_{i_1}^{[1]}, \Theta_{i_0}^{[0]}$  w.r.t. (3.12) and with pointer  $i_W$ .
37                                      $i_k \leftarrow i_k + 1$ , if  $\Theta_{i_k}^{[k]}$  was set in line 33.
38                                      $i_W \leftarrow i_W + 1$ .

```

Algorithm 3: Update the generator's data**Input :** Current value of r , and $k \in \{0, 1, 2\}$.

```

1 Calculate  $\langle r \rangle = Wr$ .
2 if  $k = 2$  then
    // Structure lists are abbreviated by
    //  $\mathcal{A} = \Theta_{\mathcal{A}}^{[2]}, \mathcal{B} = \Theta_{\mathcal{B}}^{[2]}, c = \Theta_c^{[2]}, i = \Theta_{i_W}^{[2]}$ .
3   forall  $j = 1, \dots, n_2$  do
4     if  $W_{i_j,1} > 0$  then
5        $\theta_j^{[2]} \leftarrow -(-W_{i_j,1} + \langle r \rangle_{i_j})$ .
6     else if  $\mathcal{A}_j = \mathcal{B}_j$  then
7        $\theta_j^{[2]} \leftarrow -c_j \langle r \rangle_{i_j}$ .
8     else
9        $\theta_j^{[2]} \leftarrow c_j \langle r \rangle_{i_j}$ .
10  if  $k \geq 1$  then
    // Structure lists are abbreviated by
    //  $\mathcal{A} = \Theta_{\mathcal{A}}^{[1]}, \mathcal{B} = \Theta_{\mathcal{B}}^{[1]}, c = \Theta_c^{[1]}, i = \Theta_{i_W}^{[1]}$ .
11   forall  $j = 1, \dots, n_1$  do
12     if  $i_j \neq \emptyset$  then
13       if  $W_{i_j,1} > 0$  then
14          $\theta_j^{[1]} \leftarrow 2 - W_{i_j,1} + \langle r \rangle_{i_j}$ .
15       else if  $\mathcal{A}_j = \mathcal{B}_j$  then
16          $\theta_j^{[1]} \leftarrow -c_j(c_j - 1) + c_j \langle r \rangle_{i_j}$ .
17       else
18          $\theta_j^{[1]} \leftarrow c_j \langle r \rangle_{i_j}$ .
19     else
20        $\theta_j^{[1]} \leftarrow c_j$ .
21  else
    // Structure lists are abbreviated by
    //  $\mathcal{A} = \Theta_{\mathcal{A}}^{[0]}, \mathcal{B} = \Theta_{\mathcal{B}}^{[0]}, c = \Theta_c^{[0]}, i = \Theta_{i_W}^{[0]}$ .
22   forall  $j = 1, \dots, n_0$  do
23     if  $\mathcal{A}_j = \mathcal{B}_j$  then
24        $\theta_j^{[0]} \leftarrow -\langle r \rangle_{i_j}$ .
25     else
26        $\theta_j^{[0]} \leftarrow \langle r \rangle_{i_j}$ .

```

Algorithm 4: Linear map defined by the generator**Input :** User provided vector x , population size N , and $k \in \{0, 1, 2\}$.

```

1 Initialise  $y = 0 \in \mathbb{R}^{bn}$ .
2 if  $k = 2$  then
3   forall  $i = 1, \dots, n_2$  do
4      $y_{\Theta_{i,A}^{[2]}} \leftarrow y_{\Theta_{i,A}^{[2]}} + \theta_i^{[2]} x_{\Theta_{i,B}^{[2]}}$ .
5    $y \leftarrow y/N$ .
6 if  $k \geq 1$  then
7   forall  $i = 1, \dots, n_1$  do
8      $y_{\Theta_{i,A}^{[1]}} \leftarrow y_{\Theta_{i,A}^{[1]}} + \theta_i^{[1]} x_{\Theta_{i,B}^{[1]}}$ .
9    $y \leftarrow y/N$ .
10 else
11   forall  $i = 1, \dots, n_0$  do
12      $y_{\Theta_{i,A}^{[0]}} \leftarrow y_{\Theta_{i,A}^{[0]}} + \theta_i^{[0]} x_{\Theta_{i,B}^{[0]}}$ .
13 return  $y$ 

```

3.5. From simulation to observation data

As already betokened in the preface of this chapter, in this section we will shortly describe how observation data η for the NLP (3.3)/(3.7) can be generated by using stochastic simulation. We restrict the study in this thesis to this data for three reasons: first, we have full knowledge about the inherent recombination distribution r that we want to estimate and thus can assess the results afterwards. Second, we can determine the time grid of the data as desired (and make intensively use of in the experiments in Section 3.6). And third, by now, we have no conforming real world data at hand, but discuss this issue in some more detail in the Prospect paragraph of Section 3.7.

Principle of generating observation data from simulations. Let us describe the principle idea of how to generate (artificial) observation data η_i , $i = 1, \dots, o$ of (3.3) for a given set of timepoints $\{t_1, \dots, t_o\}$ using the simulation routine of an appropriate stochastic process. Here, besides the obvious simulation of the partitioning process generated by (2.10), in principle, every tool that simulates the *coalescent-with-recombination* and returns the entire phylogenetic network, can be applied.

Remark 15. *Most of the standard tools (like ms [58]) do simulate recombination evolutionary histories, but only return the forest of embedded trees in Newick format for every site in S . It would be possible to reconstruct a directed graph from that forest by using tools (like CombineTrees [22]) which use parsimonious tree approaches; see also [99] and [2] for an overview of simulation programs of DNA sequence data with recombination. But this is not sufficient for our purpose, since this combination of tools imports an unclear*

uncertainty to our inference framework - not least because the parsimony of trees cannot correctly reproduce the bifurcations (i.e. the recombination events) of the graph.

Recently, *ARGweaver* with its subroutine *arg-sim* has been published (see [81]). The latter allows to simulate and return an entire ARG in a specific (new) fileformat *.arg. Although the ancestral recombination graph assumes the diffusion limit and therefore only a limit case of the partitioning process, in principle, it seems to be a promising alternative candidate for further studies of our approach (see also Figure 3.8 (upper part) for an illustration of the principle of producing corresponding observation data by using *arg-sim*).

The production of the ‘observation data’ η in the nonlinear least squares parameter estimation problem (3.3) from simulations is a generic two step procedure. Generic in the sense that it is applicable to any process that allows to return a partition state $\mathcal{A} \in \mathbb{P}(S)$ for a given timepoint t_i :

1. Based on the initial population’s type distribution $Z_0 = z \in E$ of (2.5) the initial (expected) normalised sampling functions

$$\mathbf{E}[H_{\Sigma_t}(z) \mid \Sigma_0 = \mathcal{A}]_{t=0} = H_{\mathcal{A}}(z) = \sum_{\mathcal{B} \succ \mathcal{A}} \frac{(N - |\mathcal{A}|)! N^{|\mathcal{B}|}}{N!} \mu(\mathcal{A}, \mathcal{B}) R_{\mathcal{B}}(z)$$

have to be evaluated $\forall \mathcal{A} \in \mathbb{P}(S)$. The assembled data matrix $(H_{\mathcal{A}}(z)^T)_{\mathcal{A} \in \mathbb{P}(S)}$ is of size $b_n \times |\mathbb{W}|$.

2. The stochastic process has to be started with all possible initial values $\Sigma_0 = \mathcal{A} \in \mathbb{P}(S)$ and stopped at a time $T > t_o$. This is repeated n_{Σ} times. The *averages of normalised sampling functions* (as approximations of the expectations) indexed by \mathcal{A} at time t_i are given by

$$\eta_i = (\eta_{i,\mathcal{A}}^T)_{\mathcal{A} \in \mathbb{P}(S)}, \quad (3.18a)$$

$$\eta_{i,\mathcal{A}} = \frac{1}{n_{\Sigma}} \sum_{k=1}^{n_{\Sigma}} H_{(\Sigma_{t_i}^{(k)} \mid \Sigma_0 = \mathcal{A})}(z) \approx \mathbf{E}[H_{\Sigma_{t_i}}(z) \mid \Sigma_0 = \mathcal{A}], \quad (3.18b)$$

where $(\Sigma_{t_i}^{(k)} \mid \Sigma_0 = \mathcal{A})$ denotes the k -th realisation (i.e. partition) of a stochastic simulation at time t_i started in $\Sigma_0 = \mathcal{A}$. Figure 3.8 provides a schematic illustration of one repetition within this procedure.

Obviously, (3.18b) is of no implementational difficulty at all, if one has the possibility to identify $(\Sigma_{t_i}^{(k)} \mid \Sigma_0 = \mathcal{A})$ and access to $H_{\mathcal{A}}(z)$. Therefore, the following paragraph gives a brief description of the algorithmic implementation of the moduls necessary to produce the initial distributions in the first step of the previous procedure. The simulation of the partitioning process is easy once the generator is available as in the previous Section 3.4.

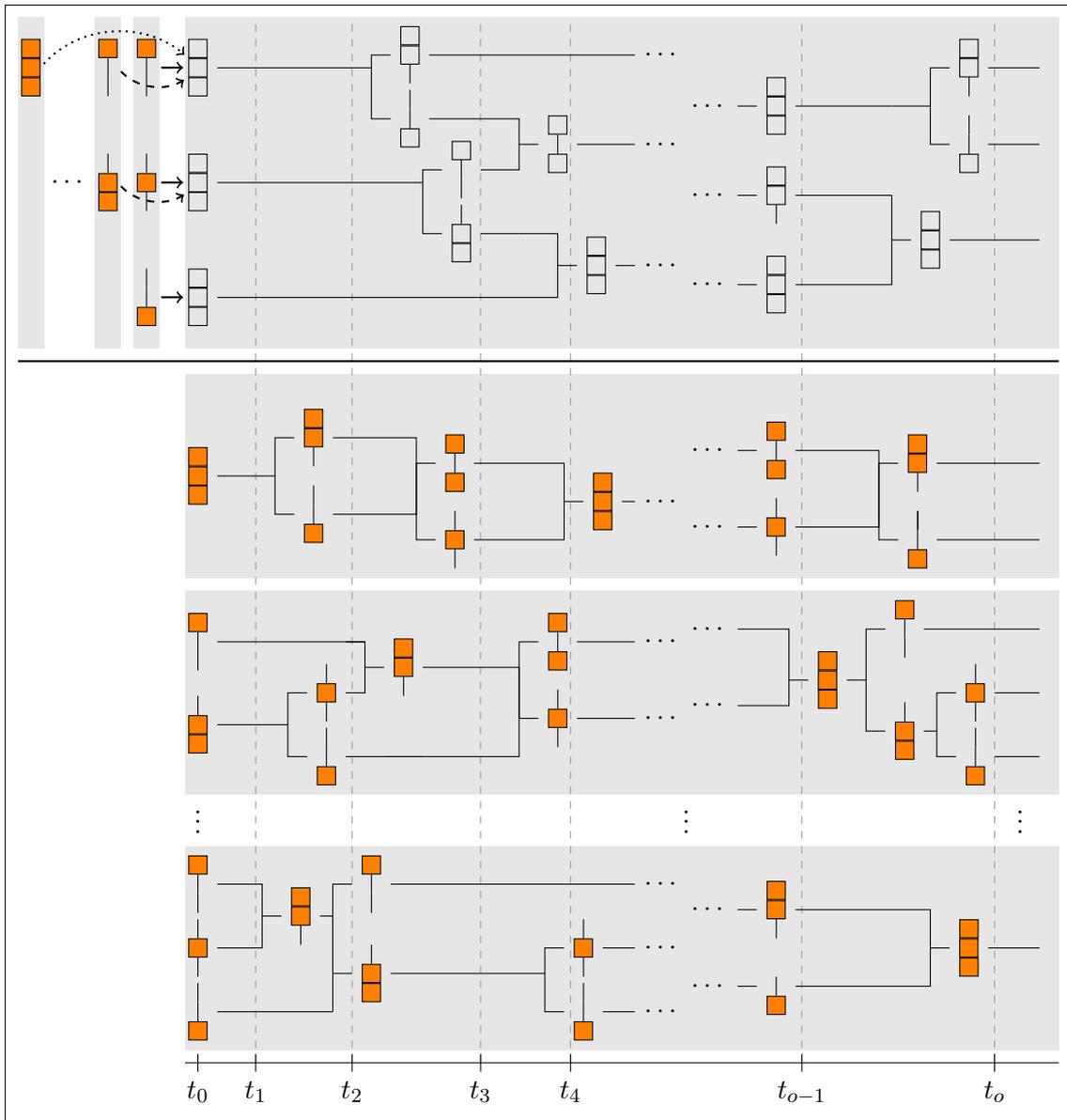


Figure 3.8.: Scheme illustrating the production of (artificial) observation data η for $n = 3$. The upper part illustrates the use of an ARG-simulation tool like `arg-sim` [81]. In this case a single realisation of the coalescent-with-recombination can be passed through for every $\mathcal{A} \in \mathbb{P}(S)$. The lower part illustrates the case of using the partitioning process characterised by (2.11). Here, for every summand in (3.18b), and in contrast to the upper case, b_n simulations (a unique one for every $\Sigma_0 = \mathcal{A}$) have to be evaluated.

Calculate initial distributions. Before we focus on some details of the algorithmical computation of the initial distributions, let us comment on the chosen representation this is based on. Apparently, we have taken (2.26), but not the equivalent (2.25) – and with good numerical cause. The latter contains products of $(N - |\mathcal{A}|)!/N! = \mathcal{O}(N^{-|\mathcal{A}|})$ with $\bar{R}_{\mathcal{B}}(z) = \mathcal{O}(N^{|\mathcal{B}|})$, which can cause enormous rounding errors, since the floating point numbers are of very different magnitude. On the contrary, recall that (2.26) multiplies normalisation-coefficients in $\mathcal{O}(N^{|\mathcal{B}| - |\mathcal{A}|})$ with relative frequencies $R_{\mathcal{B}}(z)$.

All possible and different Möbius- and normalisation-coefficients in $H_{\mathcal{A}}$ can be created in advance and independent of z , of course. Therefore, it is worthwhile to compute and store them in separate arrays

$$(\mu(\mathcal{A}, \mathcal{B}))_{\mathcal{A}, \mathcal{B}} \quad \text{and} \quad \left(\frac{(N - |\mathcal{A}|)! N^{|\mathcal{B}|}}{N!} \right)_{|\mathcal{A}|, |\mathcal{B}|}$$

of lengths

$$\sum_{\ell=1}^n \binom{n}{\ell} \quad \text{and} \quad \binom{n+1}{2},$$

respectively, in a first step. Then, (2.26) can be applied iteratively for all $\mathcal{A} \in \mathbb{P}(S)$, as soon as all marginals $\pi_{\mathcal{A}}(z)$ are available. The latter is the only slightly intricate part, for which the proposed Algorithm 5 is an iterative approach to efficiently compute the marginals $\pi_{\mathcal{A}}(z)$ for all elements \mathcal{A} of the powerset $\mathcal{P}(S)$ and a given $z \in E$.¹¹

Algorithm 5: Iterative calculation of all marginals

Input: $z \in E$

```

1 Initialise  $\pi_{S \cdot z} \leftarrow z$ ,  $J^{(0)} \leftarrow \{\emptyset\}$ 
2 for  $k = 0, 1, \dots, n-1$  do
3   while  $J^{(k)} \neq \emptyset$  do
4     Choose  $U = \arg \min_{M \in J^{(k)}} \sum_{i \in M} 2^i$ .
5     for  $j = \max U + 1, \dots, n$  do
6       Set  $V = U \cup \{j\}$  and expand  $J^{(k+1)} \leftarrow J^{(k+1)} \cup V$ .
7       // Marginalised type frequency
7        $(\pi_{S \setminus V} \cdot z)(x_{S \setminus V}) \leftarrow \sum_{\alpha \in \mathbb{X}_j} (\pi_{S \setminus U} \cdot z)(\pi_{\{i \in S \setminus U \mid i < j\}} \xi_{S \setminus U}, \alpha, \pi_{\{i \in S \setminus U \mid j < i\}} \xi_{S \setminus U})$ 
8        $j \leftarrow j + 1$ 
9      $J^{(k)} \leftarrow J^{(k)} \setminus U$ ;
10   $k \leftarrow k + 1$ 

```

Let us give a brief description of the idea Algorithm 5 is based on. The additional assumption here (and thus also in the experiments in Section 3.6) is, that the possible types $\xi \in \mathbb{X}$ allow exactly two different alleles per site (a wildtype and a mutant f.e., so $|\mathbb{X}| = 2^n$).

¹¹Recall the notations $\pi_{S \cdot z} = z$ as well as $\pi_{\emptyset \cdot z} = \|z\|$, as introduced in paragraph of Section 2.3 with the title ‘The Moran model with single-crossover recombination’, and define $\max \emptyset := 0$.

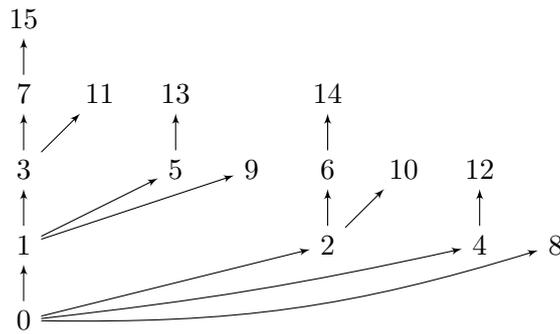


Figure 3.9.: Illustration of the hierarchy and iteration process of Algorithm 5 for $n = 4$.

Hence, note that the sum in line 7 of Algorithm 5 contains only two summands and the references to the corresponding frequencies with respect to $S \setminus U$ can be implemented by simple integer arithmetics. Furthermore, this allows to add a two-stage hierarchy to all the marginals: first, the number k of removed sites from S , and second, the canonical order of integers that have a binary representation with exactly k ones. Then, the main algorithmical idea is to do a bottom-to-top iteration with respect to the level k , i.e. going from $A = S$ ($k = 0$) to $A = \emptyset$ ($k = n$) by removing sites: for every subset A of level k , we compute all marginals of the next level by removing one of the possible remaining sites that are bigger than all removed sites of the current subset. An illustration of the hierarchy and iteration process of Algorithm 5 for $n = 4$ is given in Figure 3.9.

3.6. Numerical experiments and results

Since this paragraph shall be a feasibility study to show that the approach we have formulated up to now is capable to estimate the recombination distribution r , that determines the time course data of a population not being in stationarity, we now describe the test environment that is used for this purpose.

To give an overview right at the beginning, the entire experimental environment structure is illustrated in Figure 3.10 already and we describe the different components gradually in the following paragraphs.

Simulation scenarios (of the partitioning process) and ODE. Let us briefly describe the simulation scenarios that are considered to estimate their inherent recombination distribution with the help of our numerical approach. Such a scenario is represented by SIM and some chosen recombination distribution \mathbf{r} in Figure 3.10.

For $n \in \{2, 3, 4, 5, 6\}$, we simulate the partitioning process (Σ_t) $\bar{n}_{\Sigma_t} = 100$ times with an evolutionary time horizon of $T = 3 \cdot 10^4$. To obtain some variety in the scenarios, we choose every combination of the population size $N \in \{10^3, 10^4, 10^5\}$, the (total) probability of recombination $\mathbf{r}_{\text{tot}} \in \{1.0, 0.5, 0.2, 0.1, 0.01\}$ and the shape of the recombination distribution

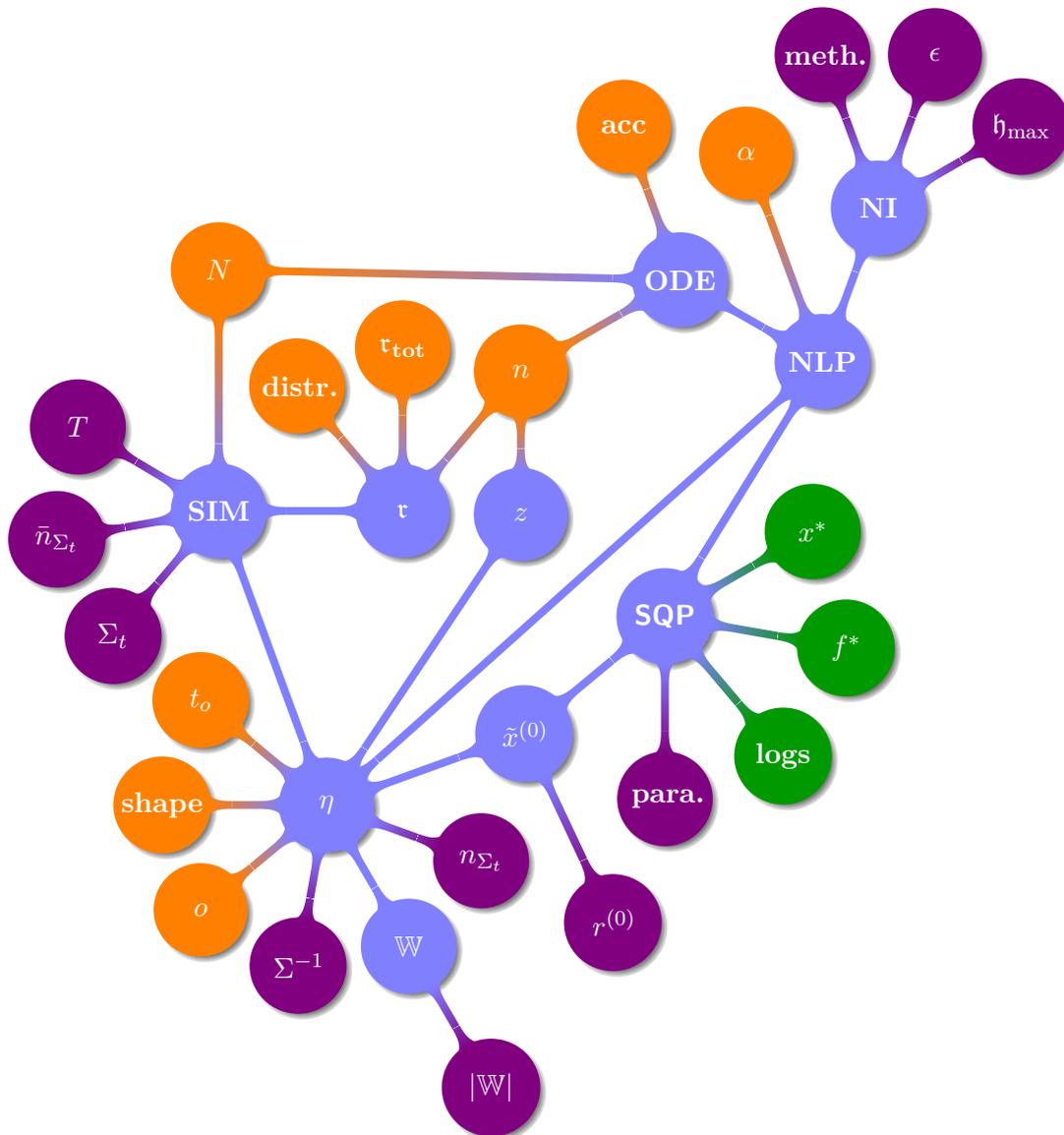


Figure 3.10.: Structure of the experimental environment. Parameters that are fixed (violet), see Table 3.1; parameters that are varied (orange), see Table 3.2; dependent components (blue); results (green). SIM means simulation scenario and NI means numerical integration module. \bullet^* means the final iterate (not necessarily at a stationary point); ‘para.’ captures Clean::SQP and Clean::ASM parameters, see Table 3.1b; ‘logs’ captures all statistics and the exit codes of SQP.

comp.	parameter	value/choice	comp.	parameter	value/choice
SIM	Σ_t	part. proc.	SQP	f mult.	1.0
	T	$3 \cdot 10^4$		overall tol.	10^{-6}
	\bar{n}_{Σ_t}	100		primal tol.	10^{-6}
η	Σ_i^{-1}	Id_{n_d}		dual tol.	10^{-6}
	$ \mathbb{W} $	1		iteration limit	39
	n_{Σ_t}	40	min penalty	10^6	
NI	method	METAN	overall tol.	10^{-6}	
	ϵ	10^{-10}	step tol.	10^{-6}	
	$\mathfrak{h}_{\max,i}$	$ t_{i+1} - t_i $	$\tilde{x}^{(0)}$	$r^{(0)}$	center

(a) Simulation scenario, observation data and numerical integration

(b) SQP parameters (subdivided in `Clean::SQP` (top five rows) and `Clean::ASM` (subsequent three rows)) and initial NLP iterate

Table 3.1.: Component settings that are *fixed* in our experimental environment.

on the set of sites to be either

$$\text{uniform:} \quad \mathfrak{r}_{\{\{1,\dots,i\}\{i+1,\dots,n\}\}} = \frac{1}{n-1} \mathfrak{r}_{\text{tot}}, \quad 1 \leq i < n, \quad (3.19a)$$

$$\text{triangular:} \quad \mathfrak{r}_{\{\{1,\dots,i\}\{i+1,\dots,n\}\}} = \begin{cases} \frac{i}{\lfloor \frac{n}{2} \rfloor (\lfloor \frac{n}{2} \rfloor + n \bmod 2)} \mathfrak{r}_{\text{tot}}, & 1 \leq i \leq \frac{n}{2}, \\ \frac{n-i}{\lfloor \frac{n}{2} \rfloor (\lfloor \frac{n}{2} \rfloor + n \bmod 2)} \mathfrak{r}_{\text{tot}}, & \frac{n}{2} < i < n, \end{cases} \quad (3.19b)$$

or, with $p = \frac{n}{2(n+1)}$,

$$\text{left geometric:} \quad \mathfrak{r}_{\{\{1,\dots,i\}\{i+1,\dots,n\}\}} = \frac{1-p}{1-p^{n-1}} p^{i-1} \mathfrak{r}_{\text{tot}}, \quad 1 \leq i < n, \quad \text{or} \quad (3.19c)$$

$$\text{right geometric:} \quad \mathfrak{r}_{\{\{1,\dots,i\}\{i+1,\dots,n\}\}} = \frac{1-p}{1-p^{n-1}} p^{n-i-1} \mathfrak{r}_{\text{tot}}, \quad 1 \leq i < n. \quad (3.19d)$$

By using all of these combinations, we obtain a scenario set of cardinality 60 for every number of sites n , albeit obviously, for $n = 2$, all the shape cases coincide, and for $n = 3$ the cases ‘uniform’ and ‘triangular’ still coincide (and thus should yield similar results in the experiments). But for $n > 3$ the scenarios are distinct, cf. Figure 3.11.

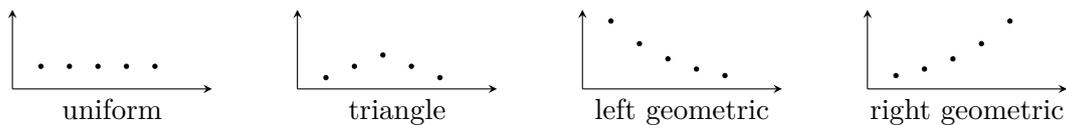


Figure 3.11.: Schematic illustration (for $n = 6$) of the different distribution shapes that are used for the simulation scenarios.

Besides, since we assume N to be known, its value, together with the number of sites, already completely determines the ODE component of our model in Figure 3.10.

Initial population (distribution) and observed type(s). As we have already announced in Section 3.5, we assume only a distinction between two alleles per site (say a wildtype and mutant, for example). This yields $|\mathbb{X}| = 2^n$. Now, in a first step, we construct a type distribution z with every $z(\xi)$ being drawn from a uniform distribution on the integers $1, \dots, 2^n$. Second, we draw $|\mathbb{W}|$ indices without replacement from the indexset $1, \dots, 2^n$ to pick the *considered types* $\xi \in \mathbb{W}$.

Note that the choice of the uniform distribution to construct z is arbitrary. Furthermore, we only consider $|\mathbb{W}| = 1$ fixed type in our experimental environment. The so computed population distribution z and the considered type ξ only depend on n . Thus z and ξ are always the same among all simulation scenarios with the same number of sites.

Observation grid and time course data. To determine an observation grid $\{t_1, \dots, t_o\}$, we set $t_1 = 0$, vary $t_o \in \{1600, 8000\}$, and vary $o \in \{6, 16\}$. The distribution (or shape) of the observation times $t_1 < \dots < t_o$ shall be either

$$\text{uniform: } t_i = (i - 1) \cdot t_o / (o - 1), \quad i = 1, \dots, o, \quad \text{or} \quad (3.20a)$$

$$\text{logarithmic: } t_i = \begin{cases} 0, & i = 1, \\ t_o^{(i-1)/(o-1)}, & i = 2, \dots, o. \end{cases} \quad (3.20b)$$

The former mimics a plain way to choose the observation times. The latter is highly adapted to the exponential solution of (2.46) with $\mathbf{E}_\varphi[H_{\mathcal{A}}(Z_0)] = H_{\mathcal{A}}(z)$. To complete the observation data component η in Figure 3.10, we choose the number of simulation runs to be averaged as $n_{\Sigma_t} = 40$ and assume $\Sigma_i^{-1} = \text{Id}_{n_d} \forall i$. Hence, we neglect (possible) information about measurement errors in this study. Figure 3.12 shows a realisation of the time course of all the (averaged) normalised sampling functions evaluated for one fixed ξ , stopped in $T = 3 \cdot 10^4$, and read on a logarithmic grid with $t_6 = 1600$.

Numerical integration and NLP/ODE adjustments. The results presented in this chapter rely on the usage of METAN as the solver for the occurring IVPs. We use default controls throughout, i.e. we make no use of passing some suitable starting step size for the next integration and control the maximal stepsize with the length of the local subinterval $\mathfrak{h}_{\max, i} = |t_{i+1} - t_i|$. The local relative error bound for the integration is set to $\epsilon = 10^{-10}$. However, we want to observe the numerical extent of the rescaling of the model parameters in (3.7) and the truncation of the generator $\Theta(r; N)$ as mentioned at the end of Section 3.4. Therefore we vary $\alpha \in \{1.0, 0.1\}$ and let the accuracy of $\Theta(r; N)$ be either *full* (application of all terms) or *reduced* (neglect all terms of the form $N^{-2}\Theta^{[2]}(r)$ in (3.11)).

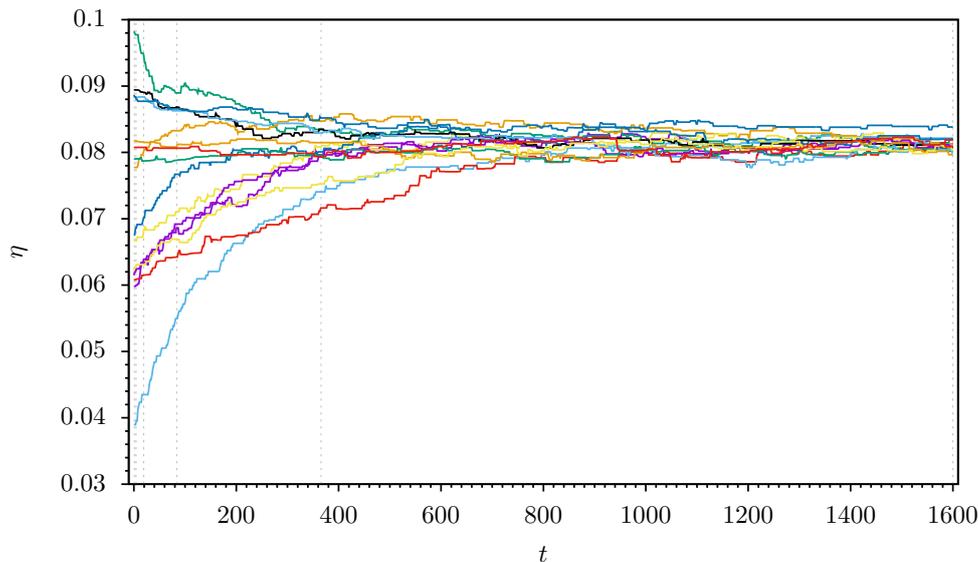


Figure 3.12.: Averaged normalised sampling functions' time courses (3.18b) for $n_{\Sigma} = 40$ realisations of the partitioning process and evaluated for one fixed type ξ . All simulations with parameters $n = 4$, $N = 10^4$, $T = 3 \cdot 10^4$, and $\tau = (0.990000, 0.006410, 0.002564, 0.001026)$ (left geometric). Observations η_i are read at $t_i \in \{0, 4, 19, 84, 366, 1600\}$ (logarithmic) for $i = 1, \dots, 6$. Plot is truncated after $t_6 = 1600$.

Set up SQP and initial NLP iterate. We use `Clean::SQP` together with the active set method `Clean::ASM` in [82] (with a regularised l_2 penalisation) as subsolver for the iterate QPs; both in version 1.0. Its sparse symmetric KKT systems are directly solved by `MA57` from the `HSL`-library (which applies a LDL^T factorisation). This combination is denoted by `SQP` in the following (and in Figure 3.10 already).

Our setting of `SQP` relies mainly on its default parameters. We have only changed the parameters mentioned in Table 3.1b. The reasoning for the choice is rather simple: it has shown good results in an early test phase on a small sample of test problems. Therefore, it has been applied unmodified to all tests that are discussed in this paragraph. Hence, the experiments are also comparable from that point of view.¹²

The composition of the initial NLP iterate is also the same throughout the experiments:

$$\tilde{x}^{(0)} = ((h^{(0)})^T, (\tilde{r}^{(0)})^T)^T = (\eta^T, \alpha(1 - n^{-1}), \alpha n^{-1}, \dots, \alpha n^{-1})^T. \quad (3.21)$$

We choose the observed data as the start iterate for the initial values in the discretisation nodes h and the center of the feasible $(n-1)$ -simplex as a start iterate for the recombination distribution r . These are also very plain choices that don't employ any further information from the origin of the data.

¹²A more sophisticated approach to arrange the `SQP` setting, best combined with the control of the numerical integration method, should come along with a future investigation. This, of course, depends on the accuracy of the data and the desired fitting accuracy of the model.

component	parameter	values
simulation scenario	n	$\{2, 3, 4, 5, 6\}$
	$\mathbf{r}_{\text{tot}}^{-1}$	$\{1, 2, 5, 10, 100\}$
	distr.	$\{\text{uniform, triangular, right geometric, left geometric}\}$
	N	$\{10^3, 10^4, 10^5\}$
grid	t_o	$\{1600, 8000\}$
	shape	$\{\text{uniform, logarithmic}\}$
	o	$\{6, 16\}$
NLP adjustment	α	$\{1.0, 0.1\}$
ODE adjustment	accuracy	$\{\text{full, reduced}\}$

Table 3.2.: Summary of parameter values determining the numerical experiments. The first four rows fix the simulation scenarios for the partitioning process. All combinations together lead to a test set of size 300. This is tested against different test cases, which are determined by a grid, that itself is specified by the rows number five to seven, together with a NLP/ODE adjustment, which is determined by the last two rows. Hereby, the parameter accuracy determines, whether all (full) or only the terms of order up to N^{-1} (reduced) of Θ are taken into account, cf. (3.11).

Numerical results. We now want to discuss the results of our numerical approach applied to *simulation scenarios* of the partitioning process. For the sake of this paragraph's readability we have collected all corresponding figures to the supplemental Appendix C. For every number of site $n \in \{2, 3, 4, 5, 6\}$, we have constructed 60 scenarios and thus have a *test set* of total cardinality 300. This is tested against the different grids and NLP/ODE adjustments as specified in Table 3.2, which will be encoded together by

$$[t_o : \text{shape} : o ; \alpha ; \text{acc}]$$

and referred to as a *test case* in the following.

By an *experiment* we denote a test case applied to an element of the test set. Then, the (*numerical*) *result* of an experiment consists of the *number of SQP iterations*, that are needed to either terminate or reach an upper runtime limit (39 iterations or 8 hours), the *final value of the objective function* $\frac{1}{2}\|h^* - \eta\|_2$, and the *distance of the final iterate* r^* to the recombination distribution \mathbf{r} (that was used in the simulations), measured by $\|r^* - \mathbf{r}\|_2 / \mathbf{r}_{\text{tot}}$.¹³ The results are illustrated in the combined Figures C.2–C.8 as tip up triangles, diamonds, and bullets, respectively. Each column, which is labelled with $\mathbf{r}_{\text{tot}}^{-1} \in \{1, 2, 5, 10, 100\}$ and separated from each other by solid a line, contains the results of the 12 experiments in which all parameters, except from N and the shape of \mathbf{r} , are fixed. See Figure C.1 for a zoom in on the column with $n = 2$, $\mathbf{r}_{\text{tot}}^{-1} = 10$ of Figure C.2.¹⁴

¹³All experiments have been executed on a desktop computer with Intel® Xeon® CPU E3-1245 v5 (3.5 GHz) and 16 GB RAM.

¹⁴The same ordering of the simulation scenarios is used in all similar figures in Appendix C.

The different colouring of the triangles, diamonds and bullets, in turn, is due to the *exit status* of SQP:

- *Green* means that either a (local) *optimal solution* is found or *the last iterate is feasible but the line search fails*.
- *Purple* means that we have a *feasible* iterate, but either the *iteration number (or time limit) is reached*.
- *Orange* means that we (still) have an *infeasible* iterate, and either the *line search fails* or the *iteration number (or time limit) is reached*.
- *Red* means that either the *solution of the local QP was impossible or another error evaluating the NLP occurred*.

And to classify the colours somewhat coarser for the purpose of this analysis: green and purple are successful experiments, orange experiments have infeasible final iterates and red experiments have been unsuccessful (mostly right from the start of the SQP solution process). Based on this coarse distinction of an experiment's result quality, we provide histograms of the exit statuses' relative frequencies with respect to the number of sites for each of the considered test cases in Figure C.10. This summary helps us to assess the degree of improvement a change of the test case parameters has on the results.

We start with the test case

$$[8000 : \text{uniform} : 6 ; 1.0 ; \text{full}],$$

and study empirically the effect of changing some of the settings on the quality of the results. From this reference case in Figure C.2 we see that for $n = 2$ at least around 60% of the experiments are successful. For $n = 4$ ($n > 4$) this ratio drops to 1.6% (0%), cf. Figure C.10. Moreover, a lot of those successful experiments yield final iterates r^* that are rather far away from the given \mathbf{r} . At least, SQP stops after very little numbers of iteration, which, on the one hand, indicates that the data seems to be of too bad quality for a reasonable model fit (fast stopping with unsuccessful exits), and, on the other hand, makes it not possible to infer the deployed recombination distribution \mathbf{r} (but, even sometimes, to identify a different local optimum). Besides, note that the very little iteration numbers also explain the echelon form of the results (bullets, for each n) regardless of the exit status. This is due to the choice of $\tilde{x}^{(0)}$ (3.21).

The first step to improve the quality of the data is to shorten the observation horizon t_o from 8000 to 1600. Hereby, the local subintervals $[t_i, t_{i+1}]$ become shorter, therefore, the extrapolation error should become smaller, and, subsequently, the evaluation of the SQP tasks (3.8) more accurate. This assumption is supported by Figure C.3 and Figure C.10, since the number of successful experiments as well as those that exit with an error (red) increases perceptibly. But, nevertheless, the overall impact of this adjustment is not very big for our test set.

A second approach is to adapt the *uniform* shape of the observation times grid to the exponential course of the type distribution trajectories, i.e. use *logarithmic* increments (3.20b), see also Figure 3.12. The massive improvement of the results is obvious in Figure C.4 even for a longer time horizon t_o compared to the previous test case. The number of successful tests increases to around 60% (average over *all* scenarios $n \in \{2, 3, 4, 5, 6\}$). Also the final iterates r^* are much closer to the deployed ones. It is remarkable, that now only those scenarios yield results with $\|r^* - \mathbf{r}\|_2 / \mathbf{r}_{\text{tot}} > 1$, which have been classified as unsuccessful (red) or infeasible with failing line search (orange) by the solver. So the data does not seem to drive the solver to local optima very different from \mathbf{r} anymore. Furthermore, now, the experiments with error exit comprises only scenarios with small \mathbf{r}_{tot} and independently of the shape of \mathbf{r} . To get an idea of the change of the time course data η depending on $\mathbf{r}_{\text{tot}}^{-1}$ and N , we give illustrations thereof restricted to the scenarios $n \in \{2, 3, 4, 5\}$ and the triangular shape of \mathbf{r} in Figures C.11–C.14.¹⁵ We conjecture, that the reason for the cumulation of unsuccessful experiments for rather big values of $\mathbf{r}_{\text{tot}}^{-1}$ may be, that in these scenarios, albeit the grid covers the exponential course quite well, the (nonstationary) dynamic is still present above rather long subintervals ($t_3 \approx 36$, $t_4 \approx 219$, $t_5 \approx 1326$), compared to smaller values of $\mathbf{r}_{\text{tot}}^{-1}$. This may cause problematic extrapolation errors. In addition, we observe from Figures C.11–C.14 that the nonstationary phase is monotonically extending with respect to n , which may be one of the reasons for the observation that the quality of the results also drops by increasing n .

As we have seen so far, both, the reduction of the time horizon t_o and the change of the grid shape increases the quality of the results, albeit to different degrees. Now we apply both adjustments at once, which yields the test case [1600:logarithmic:6;1.0;full] with the results in Figure C.5. Obviously, the shortening of the observation horizon from 8000 to 1600 improves the satisfying results of Figure C.4 even further. Apart from five (in total) infeasible experiments, all other experiments with $\|r^* - \mathbf{r}\|_2 / \mathbf{r}_{\text{tot}} > 1$ are classified as unsuccessful and detected as those already in the first iteration; in the previous test case seventeen infeasible and even six optimal experiments have such a large distance to \mathbf{r} . Now, the exit statistic in Figure C.10 reads 97% to 47% of the experiments have optimal solutions (decreasing with n) and only 2% to 25% are unsuccessful (increasing with n). In the reverse conclusion, now the data drives the solver in direction to \mathbf{r} in 98% of the not unsuccessful experiments. But also the number of unsuccessful tests increases slightly, which is compensated by a relatively bigger decrease of the number of tests in which the last iterate is still infeasible (but the line search fails or one of the runtime limits is reached). The scenario-by-scenario exit code changes from [8000:logarithmic:6;1.0;full] to [1600:logarithmic:6;1.0;full] are brought out additionally in Figure C.9. The selection of scenarios is with respect to (real) improvements or worsening (i.e. red to green / orange or orange to green, and green to orange / red or orange to red, respectively). Then, besides

¹⁵We refrain from giving the corresponding figures for $n = 6$ as well, because the $b_n = 52$ trajectories of η can not be distinguished appropriately anymore.

n	2		3		4		5		6	
	6	16	6	16	6	16	6	16	6	16
\tilde{x}	14	34	33	83	94	244	317	837	1224	3254
C_E	52	152	228	678	1504	4504	15085	45245	213156	639459
$\nabla_{\tilde{x}\tilde{x}}^2 \Lambda$	64	174	301	851	2110	6160	22037	65457	317107	948437

Table 3.3.: Number of non-zero elements for the crucial NLP components \tilde{x} , C_E and $\nabla_{\tilde{x}\tilde{x}}^2 \Lambda$.

the overall improvement regarding the number of optimal or at least feasible results, it becomes conspicuous that the change of exit statuses occurs mainly for the scenarios with $\mathbf{r}_{\text{tot}}^{-1} \neq 1, 2$. We again conjecture that this might be founded in the long subintervals ($t_3 \approx 19$, $t_4 \approx 84$, $t_5 \approx 336$, $t_6 \approx 1600$).

A third approach is to increase the number of time grid subintervals o , which may also be a remedy for the unsatisfying results of the experiments with $\mathbf{r}_{\text{tot}}^{-1} \neq 1, 2$ so far. Let us note that changing this parameter affects the NLP size (and therefore the computational costs) immediately, see Remark 11. We increase the number from $o = 6$ to $o = 16$ and get the test case [1600:logarithmic:16;1.0;full]. This changes the dimension of the primal variable \tilde{x} as well as the number of non-zero elements of the Jacobian of the equality constraints C_E and the Hessian of the Lagrangian $\nabla_{\tilde{x}\tilde{x}}^2 \Lambda$, as stated in Table 3.3. Especially, the resulting factor of around 3 in the number of non-zero entries for $n = 5, 6$ makes the total SQP solutions CPU time perceptibly longer, since the superexponential growth of b_n has a huge impact for this number of sites already (see Figure C.15). On average, the SQP computations (NLP evaluations) CPU time per iteration increases from 1.3 s (0.8 s) to 1:34.3 min (23.8 s), if we use $n = 6$ instead of $n = 5$ sites in the test case [1600:logarithmic:6;1.0;full] already. These times extend to 2.5 s (1.2 s) and 4:33.9 min (39.1 s), if we use $o = 16$ instead of $o = 6$ grid timepoints, respectively. This has to be assessed in the light of the achieved improvements illustrated in Figure C.6 in relation to Figure C.5. Concerning those, we note of the following facts:

- For $n = 2, 3, 4$ almost all experiments are solved with an optimal solution very close to \mathbf{r} in at most 11 SQP iterations.
- The only exceptions are *one* experiment with $n = 2$, which has optimal result but $\|r^* - \mathbf{r}\|_2 / \mathbf{r}_{\text{tot}} = 1.008$, *two* experiments with $n = 4$, which are feasible and reach the upper iteration limit (39) with a last iterate very close to \mathbf{r} , and *four* experiments that are unsuccessful (all again with $\mathbf{r}_{\text{tot}}^{-1} = 100$).
- Also for $n = 5$ the statistic is almost as good. 92% of the experiments have optimal solutions, *five* experiments are unsuccessful (whereof *one* even makes good progress in the first iteration but breaks off then), and only *one* experiment is (detected) infeasible after 13 iterations (albeit its final iterate r^* is close to \mathbf{r} and the objective is in the range of the other successful experiments).

- For $n = 6$ around 66% of the experiments are still successful. This is indeed significantly worse than in the experiments with fewer sites, but it is also still the best value compared to all previous test cases.

Hence, for the maximum of $n = 6$, we can conclude: the extended computational costs are conspicuous but justifiable due to the big raise of the relative frequency of optimal results in comparison to the previous test case.¹⁶

The fourth adjustment, that we want to investigate, is the impact of changing the additional parameter α , which rescales the parameter r in our modelling (3.7). We have chosen the alternative value 0.1 to produce the test case [1600:logarithmic:16;0.1;full] with the results in Figure C.7. Compared to the previous test case the amount of unsuccessful experiments could be reduced further from seventeen to four. At the same time, for $n = 5$ the amount of successful experiments unfortunately drops from 54 to 46, and the number of feasible (infeasible) experiments increases from 0 to 9 (1 to 4). Nevertheless, the final iterates r^* do not differ perceptibly from the previous test case. So this adjustment improves the quality of the results for our test set with respect to the final iterates, but somehow makes it more difficult for the solver to assess it as a local optimum in some cases.

Last, we want to examine the influence of an accuracy reduction in evaluating the ODE as suggested in Section 3.4. Therefore, the test case [1600:logarithmic:16;0.1;reduced] with its results in Figure C.8 is compared with the previous test case [1600:logarithmic:16;0.1;full]. As expected, the overall quality of the results is not worsened, since the summands omitted in (3.11) are $< 10^{-6}$. But since they make around 50% of the arithmetical operations in evaluating the system of ordinary differential equations for the tests with $n = 5, 6$ (see Table 3.4), the positive consequences on the computational CPU times as illustrated in Figure C.16 for $n = 6$ are worthwhile. We could not expect savings in the SQP computational times (upper part) per iteration, but the NLP evaluation times per iteration have been reduced in all but one experiment. Its mean value could be reduced from 32.84s to 27.29s, which is a relative improvement of around 17%.

n	n_0	n_1	n_2	saving	n	n_0	n_1	n_2	saving
5:	160	671	775	48%	6:	746	3831	5099	53%

Table 3.4.: Number of structural elements of $\Theta^{[k]}$, $k = 0, 1, 2$ and arising saving of arithmetic operations by using reduced accuracy (neglect elements of $\Theta^{[2]}$) in the cases of $n = 5, 6$.

¹⁶The situation for $n > 6$ is commented in the Prospect paragraph of Section 3.7.

3.7. Conclusion and prospect

Let us conclude the findings of this chapter. After having stated a closed system of ordinary differential equations for the expected normalised sampling functions in Chapter 2, we have formulated a NLP framework to estimate the underlying recombination distribution using a multiple shooting approach. We have shed light on an efficient algorithmical implementation of different program modules necessary to evaluate if a SQP solver is applied to find a local minimum of that NLP. In doing so, we have not only taken numerical aspects into account but also explicitly proposed appropriate datastructures and ready to use algorithms. To demonstrate the basic applicability of this approach, we have implemented all these model specific objects in C++, generated artificial observation data by simulating the partitioning process for specified recombination distributions, applied a generic SQP solver, and surveyed the quality of the results in an custom-built experimental environment for several test cases. In doing so, we have noticed that the quality of the results does not depend on the shape of the deployed recombination distribution but very crucially on the quality of the observation data; both as to be expected. Especially for a low total recombination probability τ_{tot} , a time grid with not too long subintervals, which covers the exponential course of the expected normalised sampling functions, is most suitable. A proper scaling of the parameter vector can enhance the results. Moreover, a reduction of the accuracy in evaluating the system of ordinary differential equations may save computational costs without interfering the results. However, the number of nonlinear constraints grows proportionally with the number of timepoints and superexponentially with the number of sites. This antagonises a passable computational time; especially for more than a handful of sites.

Prospect. In this first survey of the numerical approach (3.3), we have reached some (computational) limits, and made use of approximations and restrictions that may be investigated even further.

We already put a lot of effort in an efficient programming of the numerical model, see Sections 3.3–3.4. Nevertheless, already for $n > 6$ the computational costs become huge through the size of the NLP. This is due to the fast growth of the Bell number b_n . Extending Table 3.3 to the case $n = 7$ with $o = 16$ yields $\dim(\tilde{x}) = 14039$, number of non-zero elements in $C_E = 11642182$, and number of non-zero entries of $\nabla_{\tilde{x}\tilde{x}}^2 \mathcal{A} = 17411109$. This is an intractable task on the reference desktop computer that we have used for the results of Section 3.6.¹⁷ Hence, to apply this approach to more than a half-dozen sites remains challenging already. Besides, the augmentation of the differential equation with a second linear operator modelling mutations (as already done in [17]) is mathematically reasonable, but it would increase the size of the NLP and its evaluation time even further.

¹⁷Recall Footnote 13 for the reference computer’s specifications. Around 44 GB RAM are necessary now, but also no relevant SQP progress is detectable after 8 hours on a compute cluster with Intel® Xeon® CPU E5-2620 v2 (2.10GHz) (and sufficiently much RAM).

We have used a SQP algorithm to solve the NLP with an active set method as subsolver for the occurring sequence of QPs. This combination of algorithms can be applied to wide range of NLPs, since it does not make use of any special structural information about the objective function or the constraints. For a first proof of concept, this choice was sufficient. Moreover, this study served as a test problem for `Clean` (and especially for the work in [82]). Now, one adjustment in this framework could be to use another approximation or update approach for the Hessian of the Lagrangian function (as we have already noted in Remark 11). Beyond that, actually, a conventional solver for least square parameter estimation problems is a Gauß-Newton method ([73, Chap. 10], [19, Sec. 5]). Unfortunately, there was no such implementation available in `Clean` until the completion of this thesis. But the application of an appropriate Gauß-Newton method (that can handle constraints like ours) should be part of a future numerical investigation.

What is more, we only used `METAN` to solve the local IVPs. Albeit an implementation of the related `DIFSYS` was available, we refrained from reiterating all experiments with the latter. The code `METAN` applies an implicit midpoint rule and is therefore rather suited for stiff ODEs, but of higher computational costs, whereas `DIFSYS` uses an explicit approach (recall also Footnote 8). A proper eigenvalue analysis of Θ could be expedient in this context to make an assessment of the IVPs' stiffness and, therefore, to account for the reasoning of the choice for a particular numerical integration approach (whichsoever) in the end.

For a practical application of this approach on real world data, it has to be worked out how the sequenced genetic data must be processed to make it applicable to our approach, which relies on the normalised sampling functions H (2.25). And even when this task is solved, from the just mentioned computational complexity in solving the occurring NLPs, it only seems reasonable to apply this approach for a handfull of sites. But, at least for a single-cell organism with a corresponding short generation time, it may still be expedient.

Chapter 4.

Modelling and simulating Lenski's long-term evolutionary experiment

4.1. Introduction

One of the most famous instances in experimental evolution is *Lenski's long-term evolution experiment* or *LTEE* (see [65], [97], [89], [49]). Over a period of 30 years, populations of *Escherichia coli* maintained by daily serial transfer have accumulated mutations, resulting in a steady increase in fitness. The mean fitness is observed to be a concave function of time, that is, fitness increases more slowly as time goes by. In [97] a first theoretical model is formulated that builds on the underlying processes, namely mutation, selection, and genetic drift, and obtained a good agreement with the data. However, the model describes the underlying population processes in a heuristic way. As a consequence, one works with effective parameters that are hard to interpret, and it is difficult to disentangle the contributions of the various model components to the resulting fitness curve. Recently, in [47], a microscopic model is formulated for a special case (namely, for the case of deterministic fitness increments) and made explicit that the specific design of the LTEE lends itself ideally to a description via a *Cannings model* (see [31, Ch. 3.3]). In a neutral setting, this classical model of population genetics works by assigning in each time step to each of N (potential) mothers indexed $j = 1, \dots, N$ a random number ν_j of daughters such that the ν_j add up to N and are *exchangeable*, that is, they have a joint distribution that is invariant under permutations of the mother's indices. In [47], this was extended to include mutation and selection. While, in [97], they work close to the data and perform an approximate analysis in the spirit of theoretical biology, in [47] they focus on a precise definition of the model and on mathematical rigour (including, in particular, the proof of a law of large numbers in the infinite population size limit and for a suitable parameter regime).

The goal of this chapter is to build a bridge between the two approaches, to generalise the model in [47] to random fitness increments, and to also consider it in the finite-population regime. A thorough mathematical analysis will reveal the many connections between this model and the one in [97]; in particular, this will make the meaning of its parameters transparent and will allow to separate the effects of the various model

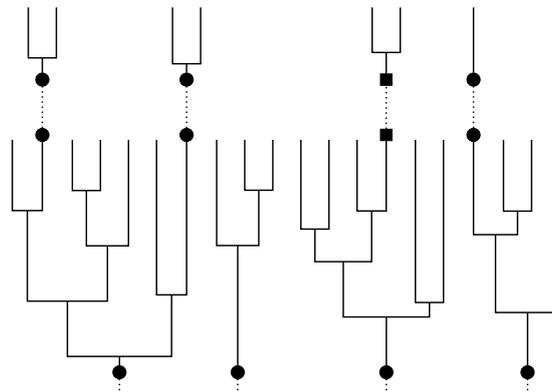


Figure 4.1.: Illustration of some day $i - 1$ (and the beginning of day i) of Lenski's LTEE with 4 founder individuals (bullets), their offspring trees within day $i - 1$, and the sampling from day $i - 1$ to i (dotted), for an average clone size of 5. The second founder from the left at day $i - 1$ (and its offspring) is lost due to the sampling, and the second founder from the right at day i carries a new beneficial mutation (indicated by the square).

ingredients. Parameter identification and stochastic simulations of a suitable extension of the model will make the connection to the experimental data. Let us briefly describe the LTEE and the outline of this chapter.

Lenski's LTEE. Every morning, Lenski's LTEE starts with a sample of $\approx 5 \cdot 10^6$ *Escherichia coli* bacteria in a defined amount of fresh minimal glucose medium. During the day (possibly after a lag phase), the bacteria divide until the nutrients are used up; this is the case when the population has reached ≈ 100 times its original size. The cells then stop dividing and enter a starvation phase. At the end of the growth period, there are therefore $\approx 5 \cdot 10^8$ bacteria, namely, $\approx 5 \cdot 10^6$ clones each of average size ≈ 100 , see Figure 4.1. The next morning, one takes a random sample of $\approx 5 \cdot 10^6$ out of the $\approx 5 \cdot 10^8$ cells, puts them into fresh medium, and the process is repeated; the sampled individuals are the roots of the new offspring trees. Note that the number of offspring a founder individual contributes to the next day is random; it is 1 on average, but can also be 0 or greater than one. Lenski started 12 replicates of the experiment in 1988, and since then it has been running without interruption. The goal of the experiment is to observe evolution in real time. Indeed, the bacteria evolve via beneficial mutations, which allow them to adapt to the environment and thus to reproduce faster. Of course neutral and deleterious mutations are more frequent than beneficial ones (see [33]), but neutral and slightly deleterious mutations will, by definition, contribute nothing or little to the adaptive process, even if they go to fixation; and strongly deleterious ones get lost quickly.

One special feature of the LTEE is that samples are frozen at regular intervals. They can be brought back to life at any time for the purpose of comparison and thus form a living

fossil record. In particular, one can, at any day i , compare the current population with the initial (day 0) population via the following *competition experiment* (see [66] and [97]). A sample from the day-0 population and one from the day- i population, each of the same size, are grown together until the nutrients are used up (say this is the case at time T_i). One then defines

$$\text{empirical relative fitness at day } i = \frac{\log(Y_i(T_i)/Y_i(0))}{\log(Y_0(T_i)/Y_0(0))}, \quad (4.1)$$

where, for $T = 0$ and $T = T_i$, $Y_i(T)$ and $Y_0(T)$ are the sizes at time T of the populations grown from the day- i sample and the day-0 sample, respectively. Note that the empirical relative fitness is a random quantity, whose outcome will vary from replicate to replicate. Figure 4.2 shows the time course over 21 years of the empirical relative fitness averaged over the replicate populations, as reported in [97].

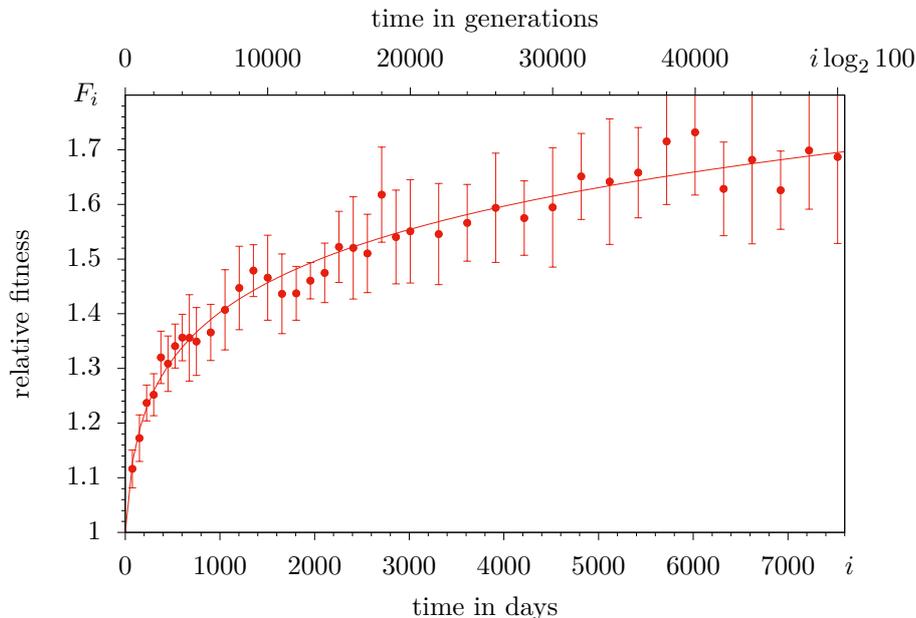


Figure 4.2.: Empirical relative fitness averaged over all 12 populations (red bullets) with error bars (95% confidence limits based on the 12 populations) from [96]; and corresponding power law (4.2) with $\hat{g} = 5.2$ and $\hat{\beta} = 5.1 \cdot 10^{-3}$ (red solid line). Data and parameters according to Fig. 2A and Table S4 of [97].

Obviously, the *mean relative fitness* has a tendency to increase, but the increase levels off, which leads to a conspicuous concave shape. As noted in [97], the mean relative fitness may be described by the power law

$$\tilde{f}(k) = (1 + \beta k)^{\frac{1}{2g}}, \quad (4.2)$$

with parameters $\beta > 0$ and $g > 0$. Here β is a time-scaling constant, and the exponent g determines the shape of the curve. Furthermore, k is time with one generation (which here

is the mean doubling time) as unit, so

$$i = \left\lfloor \frac{k}{\log_2 100} \right\rfloor \approx \frac{k}{6.6}. \quad (4.3)$$

The red solid line in Figure 4.2 shows the best fit of this curve to the data of all 12 replicate populations, as obtained in [97], with parameter estimates $\hat{g} = 5.2$ and $\hat{\beta} = 5.1 \cdot 10^{-3}$. (Here and in what follows, parameter values estimated from the data are indicated by a hat, and numbers are rounded to 2 digits. Our parameters obtained via `NonlinearModelFit` of `Wolfram Mathematica 11` only differ in the third digits.) In line with (4.1) and (4.3), we take *days* as our discrete time units, rather than doubling times (this will pay off in Section 4.2 and Section 4.3); so $\log_2 100 \approx 6.6$ generations in Figure 4.2 correspond to one day, and the total of 50000 generations correspond to around 7525 days.

The two models mentioned above aim to explain the power law (4.2). The one in [97], which we will refer to as the *WRL* model, uses an approach of *diminishing returns epistasis*, which, in turn, means that the beneficial effect of mutations decreases with increasing fitness (see, e.g., [21, p. 74] or [78]). They derive, by partly heuristic methods, a differential equation for the mean relative fitness whose solution is given by (4.2). The time-scaling parameter β is determined by the interplay of the rate and the effect of beneficial mutations, with the heuristics in [42] for the description of *clonal interference*¹⁸ playing an important role. The second approach is the individual-based model in [47] and makes full use of ideas, concepts, and techniques from mathematical population genetics, which seem to be ideally tailored for the LTEE setup. We will address this as the *GKWY* model; since it has been published in a mathematical journal, we will review it in more detail in Section 4.2 with an emphasis on the biological content. For a certain parameter regime that excludes clonal interference, and using a similar approach to diminishing returns as in the WRL model, in [47] a law of large numbers (as $N \rightarrow \infty$) is proved, thereby rigorously deriving a version of the power law (4.2).

Goal and outline. A major goal of this chapter is to provide a thorough mathematical model of the LTEE, and to relate it to the observed fitness curve via parameter estimation and stochastic simulations. This approach will provide additional connections between the ideas contained in the WRL and the GKWY model addressed in the previous paragraph. The design of the LTEE, with the daily growth cycles and the sampling scheme, results in an (approximately) constant population size at the beginning of each day. As made explicit in [47], this lends itself in a prominent way to a description through a Cannings model (including mutation and selection), where the mothers are identified with the founders in a given day and the daughters with the founders in the next day. The crucial parameter of

¹⁸Clonal interference (see [42], [41], [76]) refers to the situation of two (or more) beneficial mutations present in the population at the same time. They then compete with each other and, in the end, only one of them will be established in the population; an effect that slows down adaption (when measured against the stream of incoming mutations), and biases the distribution of beneficial effects.

the Cannings model, namely, the *variance of the number of offspring* of a founder individual that make it into the next day, is obtained in the context of the LTEE from an explicit stochastic model of population growth during each day. This offspring variance enters Haldane's formula for the fixation probability, see (4.17) below.

As a matter of fact, also in [97] a formula for the fixation probability is used (see Eq. (S1) in their Supplementary Text). In this context they refer to [42], which assumes a deterministic population growth (and clones of equal size) resulting from synchronous divisions. Indeed, the Cannings model which thus is implicitly hidden in the WRL model, turns out to work with a different offspring variance; we will come back to this in Section 4.5.

In addition to the specification of the offspring variance, our model for the daily population growth in continuous time allows us to quantify selection (including diminishing returns epistasis) at the level of the individual reproduction rates within a day. The effect of diminishing returns seems to be obvious from Figure 4.2; however, epistasis is not the only contribution to the fitness curve. Rather, the design of the experiment also has its share in it via what we call the *runtime effect*, namely, the shortening of the daily growth phase with increasing fitness. The analysis of our model will allow a clear separation of these contributions. Likewise, the population-genetic notions that also appear in the WRL model (namely, the mutation rate, the selective advantage, the effective population size, the fixation probability, and the strength of epistasis) will be made precise in terms of the underlying microscopic model. Throughout, we aim at a rigorous mathematical treatment where possible.

The chapter is organised as follows. In Section 4.2, we will recapitulate the GKWY model and explain its law of large numbers (that is a deterministic limit in a suitable parameter regime as population size goes to infinity) for a more biological readership. At the end of Section 4.2, we will consider the resulting stochastic effects in a system whose parameters are obtained from a fit to the data observed in the LTEE (and which thus naturally differs from its infinite population limit). In Section 4.3, this will lead us to consider clonal interference, which is present if the population size is finite in our model. We will investigate clonal interference both for the case of deterministic and random fitness increments. Here we do not prove a law of large numbers, but derive approximations with the help of moment closure and a refined version of the Gerrish-Lenski heuristics. For each case of fitness effects we estimate relevant parameters and enhance the plausibility of our modelling with stochastic simulation studies by applying suitable algorithms as stated in Section 4.4. Then, in Section 4.5, we will thoroughly discuss various modelling aspects in the context of both the WRL and the GKWY models, in particular the notions of fitness increment, selective advantage, and epistasis, as well as the equivalent concepts of offspring variance, pair coalescence probability, and effective population size.

4.2. A probabilistic model for the LTEE and its law of large numbers

The GKWY model takes into account two different dynamics, namely, the dynamics *within each individual day*, and the dynamics *from day to day*, together with a suitable *scaling regime*. The resulting *relative fitness process* is proved to converge, in the $N \rightarrow \infty$ limit, to a power law equivalent to (4.2); that is, the power law arises as a *law of large numbers*. We explain this here with the help of an appropriate *heuristics*. In what follows, we present these building blocks and perform a first *reality check*.

Intraday dynamics. Let T be (continuous) physical time within a day, with $T = 0$ corresponding to the beginning of the growth phase (that is, we discount the lag phase). Day i starts with N founder individuals ($N \approx 5 \cdot 10^6$ in the experiment). The reproduction rate (or *Malthusian fitness*) of founder individual j at day i is R_{ij} , $0 \leq i, 1 \leq j \leq N$. It is assumed that at day 0 all individuals have identical rates, $R_{0j} \equiv R_0$, so the population is *homogeneous*. Offspring inherit the reproduction rates from their parents.

We use dimensionless variables right away. Therefore we denote by

$$t = R_0 T \quad \text{and} \quad (4.4)$$

$$r_{ij} = \frac{R_{ij}}{R_0} \quad (4.5)$$

dimensionless time and rates, so that on the time scale t there is, on average, one split per time unit at the beginning of the experiment (this unit is 55 minutes, cf. [11]) and $r_{0j} \equiv 1$. In this paragraph, we consider the r_{ij} as given (non-random) numbers.

We thus have N independent *Yule processes* at day i : all descendants of founder individual j (the members of the j -clone) branch at rate r_{ij} , independently of each other. They do so until $t = \sigma_i$, where σ_i is the duration of the growth phase on day i . We define σ_i as the value of t that satisfies

$$\mathbf{E}[\text{population size at time } t] = \sum_{j=1}^N e^{r_{ij} t} = \gamma N, \quad (4.6)$$

where γ is, equivalently, the multiplication factor of the population within a day, the average clone size, and the dilution factor from day to day in the experiment ($\gamma \approx 100$ in the LTEE). Note that the Yule processes are stochastic, so the population size at time t is, in fact, random; in the definition of σ_i , we have idealised by replacing this random quantity by its expectation. Since N is very large, this is well justified, because the fluctuations of the random time needed to grow to a factor 100 in size are small relative to its expectation.

Interday dynamics. At the beginning of day $i > 0$, one samples N new founder individuals out of the γN cells from the population at the end of day $i - 1$. We assume

that one of these new founders carries a *beneficial mutation* with probability μ_N ; otherwise (with probability $1 - \mu_N$), there is no beneficial mutation. We think of μ_N as the probability that a beneficial mutation occurs in the course of day $i - 1$ and is sampled for day i .

Assume that the new beneficial mutation at day i appears in individual m , and that the reproduction rate of the corresponding founder individual k in the morning of day $i - 1$ has been $r_{i-1,k}$. The new mutant's reproduction rate is then assumed to be

$$r_{im} = r_{i-1,k} + \delta(r_{i-1,k}) \text{ with } \delta(r) := \frac{\varphi_N}{r^q}. \quad (4.7)$$

Here, φ_N is the beneficial effect due to the first mutation (that is $\delta(1)$, which applies while $r = 1$), and q determines the strength of epistasis. In particular, $q = 0$ implies constant increments (that is, additive fitness), whereas $q > 0$ means that the increment decreases with r , that is, we have diminishing returns epistasis. Note that, at this stage, the fitness increment is a *deterministic* function of the mother's reproduction rate. This is in line with the *staircase model* of population genetics (see [36], [24]). We will turn to stochastic increments in Section 4.3.

Let M_i^N be the number of new mutants at the beginning of the day. So far we have assumed that M_i^N can only take the values 1 or 0. More generally, for describing the random number of individuals that are offspring of new mutants from day $i - 1$ and make it into the N -sample at the beginning of day i , we might consider integer-valued random variables M_i^N with small expectation μ_N . We assume that the distribution of M_i^N does not depend on the current fitness value, and, as in (4.7), that any mutation adds $\delta(r) = \varphi_N/r^q$ to the pre-mutant reproduction rate. As long as μ_N is not very small, precision may be added by using Poisson random variables, which is what we do in the simulations, see Section 4.4. One might also think of a finer *intraday modelling* of the mutation mechanism, cf. [93] or [64]. However, although the limit theorem in [47] is proved only for binary random variables M_i^N , we conjecture that its assertion also holds for non-binary random variables M_i^N in the scaling regime (4.10) discussed below, at least as long as the variances of the M_i^N remain bounded as $N \rightarrow \infty$. We will adhere to the binary assumption in our analysis, and it will turn out as an excellent approximation. Note also that we have idealised by not taking into account the change in fitness due to mutation during the day; this is because a mutant appearing during the day will not rise to appreciable frequency in the course of this first day of its existence, and thus will not change the overall growth rate of the population in any meaningful way.

Mean relative fitness. With a view towards (4.1) we define the *mean relative fitness*, depending on the configuration of reproduction rates r_{ij} of the N individuals in the sample at the beginning of day i , as

$$F_i^N := \frac{1}{\sigma_i} \log \left(\frac{1}{N} \sum_{j=1}^N e^{r_{ij}\sigma_i} \right). \quad (4.8)$$

Here, σ_i is as defined in (4.6). Comparing (4.1) and (4.8), we see that the former contains additional sources of randomness: on the one hand, the numerator of (4.1) may be viewed as stemming from a sample that was drawn from the population at the end of day $i - 1$ (and which consists of individuals different from those present at the beginning of day i), on the other hand the duration of the growth phase leading to (4.1) is not a predicted time as in (4.8) but an empirical time coming out of the competition experiment between the samples from day i and day 0. However, since the samples consist of a large number of individuals, the random variables occurring in (4.1) will, with high probability, come out close to their expectations, thus making already a single copy of the random variable (4.1) a reasonably good approximation of (4.8), at least if the population at day i is sufficiently homogeneous.¹⁹ Note that (4.8) implies that

$$e^{F_i^N \sigma_i} = \frac{1}{N} \sum_{j=1}^N e^{r_{ij} \sigma_i}. \quad (4.9)$$

Thus, F_i^N may be understood as the *effective reproduction rate* of the population at day i , which is different from the mean Malthusian fitness $\frac{1}{N} \sum_j r_{ij}$ unless the population is homogeneous, that is, $r_{ij} \equiv r_i$.

Scaling regime. We have indexed μ_N and φ_N with population size because the law of large numbers requires to consider a sequence of processes indexed with N and to take the limit $N \rightarrow \infty$. More precisely, we will take a *weak mutation — moderate selection limit*, which requires that μ_N and φ_N become small in some controlled way as N goes to infinity. Specifically, in [47] it is assumed, that

$$\mu_N \sim \frac{1}{Na}, \quad \varphi_N \sim \frac{1}{Nb} \text{ as } N \rightarrow \infty, \text{ with } 0 < b < \frac{1}{2}, \quad 3b < a. \quad (4.10)$$

This entails that φ_N is of order greater than $1/\sqrt{N}$ but of order less than 1. Due to the assumption $a > 3b$, μ_N is of much lower order than φ_N . This is used in [47] to prove that, as $N \rightarrow \infty$, with high probability no more than two fitness classes are simultaneously present in the population over a long time span. Note that μ_N is the *per-day mutation probability per population* (but see the discussion at the end of the paragraph on interday dynamics).

Furthermore, the scaling of φ_N implies that selection is stronger than genetic drift as soon as the mutant has reached an appreciable frequency. The method of proof applied in [47] requires the assumptions (4.10) in order to guarantee a coupling between the new

¹⁹Indeed, due to the enhanced reproduction rates at day i compared to day 0, the nutrient consumption time T_i in the experiment leading to (4.1) will generically be longer than σ_i , the predicted nutrient consumption time in (4.8). This is because T_i refers to a mixture of day- i and day-0 populations (of equal size), whereas σ_i relates to a ‘pure’ day- i population. If the day- i population is homogeneous, this effect will cancel out; but if it is inhomogeneous, (4.1) will be systematically larger than (4.8), because then the individuals with a larger reproduction rate will get more weight in (4.1) than in (4.8).

mutant's offspring and two nearly critical Galton-Watson processes between which the mutant offspring's size is 'sandwiched' for sufficiently many days. Specifically, under the assumption $0 < b < \frac{1}{2}$, the coupling applied in [47] works until the mutant offspring in our Cannings model has reached a small (but strictly positive) proportion of the population, or has disappeared. A careful inspection of the arguments shows that, under the weaker condition $0 < b < \frac{2}{3}$, this coupling works at least until the mutant offspring has (either disappeared or) reached size N^b , from which it then goes to fixation by a law of large numbers argument. This makes the limit result in [47] valid for $0 < b < \frac{2}{3}$; we conjecture that it even holds for $0 < b < 1$.

In the case where selection is much stronger than mutation, the classical models of population genetics, such as the Wright-Fisher or Moran model, display a well-known dynamics. Two distinct scenarios can happen, see, e.g., [53, Chap. 2 and Fig. 2.7]: either a fast loss of a new beneficial mutation, or its fixation. We will see that qualitatively our Cannings model displays a quite similar behaviour. Furthermore, as already indicated, with the chosen scaling the population turns out to be homogeneous on generic days i as $N \rightarrow \infty$. This has the following practical consequences for the *relative fitness process* $(F_i^N)_{i \geq 0}$ defined by (4.8). First, on a time scale with a unit of $1/(\mu_N \varphi_N)$ days, $(F_i^N)_{i \geq 0}$ turns into a jump process as $N \rightarrow \infty$, cf. Figure 4.3. Second, on the (generic) days i at which the populations are nearly homogeneous, the subtle systematic difference between (4.1) and (4.8), as described in Footnote 19, will disappear.

Heuristics leading to the limit law. Assume a new mutation arrives in a *homogeneous* population of relative fitness F . It conveys to the mutant individual a relative *fitness increment*

$$\delta_N(F) = \frac{\varphi_N}{F^q}, \quad (4.11)$$

that is, the mutant has relative Malthusian fitness $F + \delta_N(F)$. The length of the growth period then is

$$\sigma(F) = \frac{\log \gamma}{F} \quad (4.12)$$

(since this solves $e^{Ft} = \gamma$, cf. (4.6)). We now define the *selective advantage* of the mutant as

$$s_N(F) = \delta_N(F) \sigma(F). \quad (4.13)$$

Obviously, *the length σ of the growth period decreases with increasing F* and, since s_N in (4.13) decreases with decreasing σ , s_N would decrease with increasing F even if $\delta_N(F)$ were constant. This is what we call the *runtime effect*: adding a constant to an interest rate F of a savings account becomes less efficient when the runtime decreases.

Let us explain the reasoning behind (4.13). In population genetics, the selective advantage (of a mutant over a wildtype) per generation is

$$s = \frac{a_1 - a_0}{a_0}, \quad (4.14)$$

where a_0 (a_1) is the expected number of descendants of a wildtype (mutant) individual in one generation; Eq. (4.14) has the form of a *return* (of a savings account, say). If growth is in continuous time with Malthusian parameters r_0 and $r_1 = r_0 + \delta$, respectively, and a generation takes time σ , then $a_0 = e^{r_0\sigma}$ and $a_1 = e^{r_1\sigma} \approx a_0(1 + \delta\sigma)$ if δ is small, which turns (4.14) into (4.13). Often, the appropriate notion of a generation is the time until the population has doubled in size, see e.g. Eq. (3.2) in [23], which provides an analogue to (4.13). In our setting, the corresponding quantity is the time required for the population to grow to γ times its original size, which is the length $\sigma(F)$ of the growth period in (4.12).²⁰ Together with the above expression for s , this explains (4.13). Notably, a formula that is perfectly analogous to (4.13) also appears in [85, p. 1977, last line]; there, the concept of a viral generation is associated with the cell infection cycle, and the number K (which corresponds to our γ) is the burst size or viral yield per cell.

Furthermore, it is precisely this notion of selection advantage conveyed by (4.13) and (4.14) that governs the *fixation probability*. Namely, the fixation probability of the mutant turns out to be

$$\pi_N(F) \sim C s_N(F). \quad (4.15)$$

Here, \sim means asymptotic equality²¹, and $C := \gamma/(\gamma - 1)$ is asymptotically twice the reciprocal offspring variance in one Cannings generation of the GKWY model²²; that is, with the notation introduced in the first paragraph of the Introduction, the offspring variance v in one Cannings generation satisfies

$$v = \mathbf{E}[(\nu_1 - \mathbf{E}[\nu_1])^2] \sim 2 \frac{\gamma - 1}{\gamma} = \frac{2}{C}. \quad (4.16)$$

Hence (4.15) is in line with Haldane's formula

$$\pi \sim \frac{s}{v/2}, \quad (4.17)$$

which says that the fixation probability π is (asymptotically) the selective advantage s divided by half the offspring variance v in one generation. Haldane's formula relies on a branching process approximation of the initial phase of the mutant growth; see [77] for an account of this method, including a historic overview.

For the sake of completeness, let us give the following intuitive explanation for (4.16). In

²⁰In line with this, we choose days as our discrete time units, as already mentioned in Section 4.1.

²¹That is, $\pi_N(F)/(C s_N(F)) \rightarrow 1$ as $N \rightarrow \infty$, see [47].

²²Let us emphasise once again that one generation of this Cannings model corresponds to one day in the LTEE.

every Cannings model, one has the relation

$$v = (N - 1)p_{\text{coal}} \quad (4.18)$$

between v and the pair coalescence probability p_{coal} , that is, the probability that two randomly sampled daughters have the same mother, cf. [26, Ch. 4.1]. Eq. (4.18) then follows readily from the elementary relation

$$p_{\text{coal}} = \mathbf{E}\left[\frac{1}{N(N-1)} \sum_j \nu_j(\nu_j - 1)\right] = \frac{1}{N-1}(\mathbf{E}[\nu_1^2] - 1) = \frac{1}{N-1}v,$$

because the ν_j are exchangeable and sum to N by assumption. In our specific Cannings model, the family size of a randomly sampled daughter individual at the end of the day is, on average, asymptotically twice as large as a typical family size.²³ Since we have N clones of average size γ , and the sampling is without replacement, we have

$$p_{\text{coal}} \sim \frac{2}{N} \frac{\gamma - 1}{\gamma}. \quad (4.19)$$

Together with (4.18) this implies (4.16). Note that (4.19) at the same time defines the (coalescence) effective population size via $N_e = 1/p_{\text{coal}}$, cf. [31, Ch. 3.7] or [26, Ch. 4.4]. Another crucial ingredient of the heuristics is the time window of length

$$u_N(F) = \frac{\log N}{s_N(F)} \quad (4.20)$$

after the appearance of a beneficial mutation that will survive drift (a so-called *contending mutation*) in the fitness background F ; this approximates the expected time it takes for the mutation to become dominant in the population. Indeed, (4.20) is asymptotically equivalent to the solution of

$$(1 + s_N)^i = \varepsilon N$$

for any positive constant ε ; here, the left-hand side is the expected offspring size of a mutant after i days in the branching process approximation, and the right-hand side is a sizeable proportion of the population.

All this now leads us to the dynamics of the relative fitness process. As illustrated in

²³The size of the clone to which a sampled individual belongs has a *size-biased* distribution; this is in line with the classical *waiting time paradox* (cf. [40, Example 4.16]). In our model, the size distribution of a typical clone at the end of the day is approximately geometric with parameter $1/\gamma$, and the size-biasing of this distribution is (approximately) negative binomial with parameters 2 and $1/\gamma$. Consequently, the expected size of the clone to which a sampled individual belongs is approximately 2γ , that is twice the expected size of a typical clone. This proportion carries over from the clones to the families of sampled individuals. Let us emphasise once again that a *family* consists of the founders at the beginning of the next day that go back to the same founder in the current day; whereas a *clone* consists of all descendants of a founder at the end of a day, regardless of whether they are sampled for the next day or not.

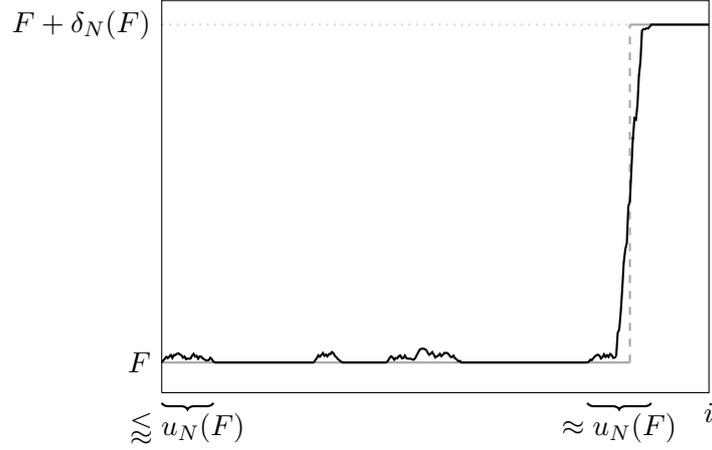


Figure 4.3.: The relative fitness process (black) and the approximating jump process (grey).

Figure 4.3, most mutants only grow to small frequencies and are then lost again (due to the sampling step). But if it does happen that a mutation survives the initial fluctuations and gains appreciable frequency, then the dynamics turns into an asymptotically deterministic one and takes the mutation to fixation quickly, cf. [26, Ch. 6.1.3]. Indeed, within time $u_N(F)$, the mutation has either disappeared or gone close to fixation; by (4.10), this time is much shorter than the mean interarrival time $1/\mu_N$ between successive beneficial mutations. As a consequence, there are, with high probability, at most two types present in the population at any given time (namely, the *resident* and the *mutant*), and *clonal interference is absent*. Therefore, in the scenario considered, survival of drift is equivalent to fixation. Next, we consider the expected per-day increase in relative fitness, given the current value F . This is

$$\mathbf{E}[\Delta_N F \mid F] \approx \mu_N \pi_N(F) \delta_N(F) \sim \frac{\Gamma_N}{F^{2q+1}}. \quad (4.21)$$

Here, the asymptotic equality is due to (4.11)–(4.13) and (4.15), and the compound parameter

$$\Gamma_N := C\mu_N\varphi_N^2 \log \gamma \quad (4.22)$$

is the rate of fitness increase per day at day 0 (where $F_0 \equiv r_0 = 1$). Note that φ_N/F^q appears squared in the asymptotic equality in (4.21) since it enters both π_N and δ_N . Note also that the additional +1 in the exponent of F comes from the factor of $1/F$ in the length of the growth period (4.12), and thus reflects the runtime effect.

This now leads us to define a new time variable τ related to i of (4.3) via

$$i = \left\lfloor \frac{\tau}{\Gamma_N} \right\rfloor \quad (4.23)$$

with Γ_N of (4.22), which means that one unit of time τ corresponds to Γ_N days. With this rescaling, we have

$$F_{[\tau/\Gamma_N]}^N \rightarrow f(\tau) \text{ as } N \rightarrow \infty,$$

where f satisfies the initial value problem

$$\frac{d}{d\tau} f(\tau) = \frac{1}{f^{2q+1}(\tau)}, \quad f(0) = 1, \quad (4.24)$$

with solution

$$f(\tau) = (1 + 2(1 + q)\tau)^{\frac{1}{2(1+q)}}. \quad (4.25)$$

Note that (4.24) is just a rescaling limit of (4.21), where the expectation was omitted due to the scaling regime applied, as will be explained next.

Law of large numbers. The precise formulation of the limit law in [47] reads

Theorem 2. *For $N \rightarrow \infty$ and under the scaling (4.10), the sequence of processes $(F_{[\tau/\Gamma_N]}^N)_{\tau \geq 0}$ converges in distribution and locally uniformly, to the deterministic function $(f(\tau))_{\tau \geq 0}$ in (4.25).*

The theorem was proved along the heuristics outlined above²⁴ with the help of advanced tools from probability theory. It is a law of large numbers reasoning, which allows to go from (4.21) to (4.24) (and thus to ‘sweep the expectation under the carpet’), in the sense that, for large N and under the scaling assumption (4.10), fitness is the sum of a large number of small per-day increments accumulated over many days, and may be approximated by its expectation.

Since time has been rescaled via (4.23), Eq. (4.25) has q as its single parameter. Note that $1/(2(1 + q)) < 1$ (leading to a concave f) whenever $q \geq 0$; in particular, *the fitness curve is concave even for $q = 0$, that is, in the absence of epistasis*. This is due to the runtime effect: if the population as a whole already reproduces rather fast, then the end of the growth phase is reached sooner and thus leaves less time for a mutant to play out its advantage; see also the discussion in Section 4.5. The second parameter, namely Γ_N , reappears when τ is translated back into days; that is, $F_i^N \approx f(\Gamma_N i)$. Note that R_0 , as used in the first nondimensionalisation step (4.4), is not an additional parameter because it is already absorbed in φ_N^2 .

A first reality check. The limit law (4.25) is identical with the power law (4.2) in [97] up to a transformation of the parameters that relies on relevant details in the modelling (see also the discussion in Section 4.5). We have $q = g - 1$, so $\hat{g} = 5.2$ of Section 4.1 translates into $\hat{q} = 4.2$.²⁵ Furthermore, $\Gamma = \beta \log_2(\gamma)/(2(1 + q))$ due to (4.2) and (4.25) together

²⁴Note that in [47] they partly work with dimensioned variables, which is why the notation and the result look somewhat different.

²⁵Recall that we denote parameter estimates by a hat to distinguish them from the corresponding theoretical quantities. Figures are rounded to two digits.

with the fact that $k = \tau \log_2(\gamma)/\Gamma$ by (4.3) and (4.23); given $\hat{\beta} = 5.1 \cdot 10^{-3}$, this results in $\hat{\Gamma} = 3.2 \cdot 10^{-3}$ (here and in what follows, we suppress the index N , since we will work with fixed, finite N from now on). The resulting fit is reproduced in Figure 4.4 (red solid

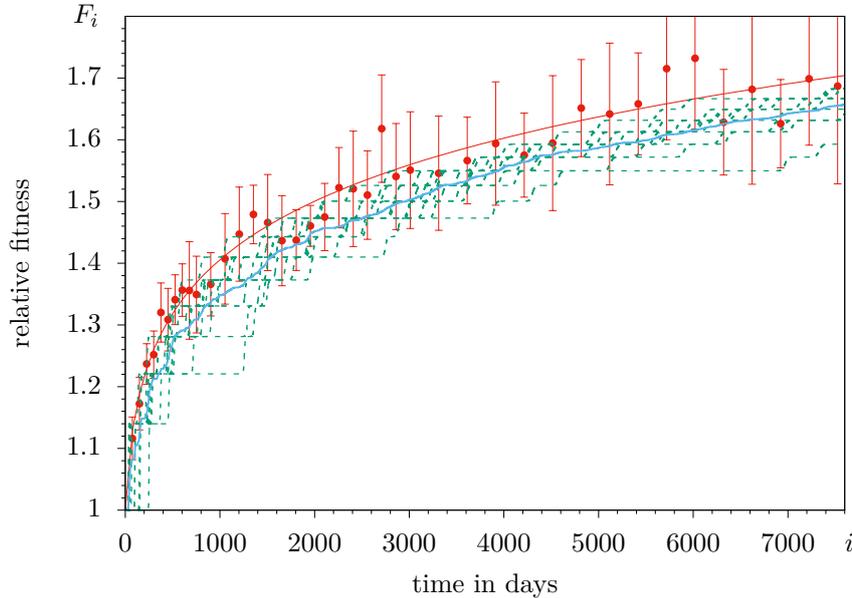


Figure 4.4.: Least-squares fit of the curve (4.25) to the data in [96], and stochastic simulations of finite populations with deterministic beneficial effects. Red bullets: mean empirical relative fitness (averaged over all 12 populations) with error bars as in Figure 4.2; solid red line: $F_i \approx f(\hat{\Gamma}i)$ with parameter values $\hat{q} = 4.2$ and $\hat{\Gamma} = 3.2 \cdot 10^{-3}$; green lines: 12 individual trajectories F_i obtained via Cannings simulations with $N = 5 \cdot 10^6$, $\gamma = 100$, $\hat{\varphi} = 0.14$, and $\hat{\mu} = 0.035$; light blue line: average over the 12 simulations.

line). In line with [97, Fig. 2], we average over all 12 populations, at this point neglecting a certain variability of the parameters between the populations, see their Table S4.

In the light of (4.22), of the given value $\hat{\Gamma}$, and of the fact that $C \log \gamma \approx 4.7$, the values of $\hat{\mu}$ and $\hat{\varphi}$ cannot both be very small. We therefore now check the limit law against realistic parameter values. We start by decomposing the compound parameter Γ . Recall from (4.7) that the *fitness increment due to the first beneficial mutation* is

$$\varphi = \delta(F_0) = \delta(1). \quad (4.26)$$

This was estimated as 0.1 in [65] (see also [42], and [97]). For reasons to be explained in Section 4.3, however, we work with the somewhat larger value $\hat{\varphi} = 0.14$. The mutation probability may then be obtained from (4.22) as

$$\hat{\mu} = \frac{\hat{\Gamma}}{C \hat{\varphi}^2 \log \gamma} = 0.035. \quad (4.27)$$

Stochastic simulations of the GKWY model, performed with Algorithm 6 described in Section 4.4 and using the above parameters together with $N = 5 \cdot 10^6$, are also shown in Figure 4.4. Their mean (over 12 runs) recovers the basic shape of the fitness curve, but systematically underestimates both the limit law and the data. A natural explanation for this is clonal interference, which is absent in the limit under the scaling (4.10), but leads to loss of mutations for finite N . This will be taken into account in Section 4.3. But let us note here that the fluctuations in the data are rather larger than those of the simulations; this may well go along with a variability of the parameters between the 12 replicates of the LTEE, which is present in the data, but not in our simulations.

4.3. Including clonal interference

As discussed in Section 4.2, the scaling regime in the GKWY model was such that, with high probability, no new beneficial mutation arrived while the previous one was on its way either to extinction or fixation. As indicated by the simulation results in Figure 4.4, also clonal interference should be taken into account. Briefly stated, clonal interference refers to the situation where a second contending mutation appears while the previous one is still on its way to fixation (recall also Footnote 18). It is crucial to keep in mind that, unlike the case without clonal interference considered in Section 4.2, survival of drift may then no longer be identified with fixation; rather, there may be an additional loss of contending mutations due to clonal interference. In particular, the quantity π of (4.15) must now be addressed as the *probability to survive drift* rather than the fixation probability.

A full analytic treatment of clonal interference is beyond the scope of this section; in particular, we will not prove a law of large numbers here. Rather, we refine and adapt the heuristics of [42], see also [97]. The heuristics was originally formulated for fitness effects that follow an exponential distribution. We will, however, first consider the deterministic effects as assumed in the GKWY model and then proceed to random effects from a very general probability distribution.

Deterministic beneficial effects. For the case of deterministic beneficial effects, we will sketch and apply a *thinning heuristics*, as a counterpart of the heuristics in [42]. Consider the situation that a second mutation surviving drift appears within the time window $u(F) := u_N(F)$ of (4.20) after the appearance of a first mutation (this is more or less while the first mutation has not become dominant yet). Then, with high probability, the second mutation occurs in an individual of relative fitness F (rather than in an individual of relative fitness $F + \delta(F)$), and therefore belongs to the same fitness class as the first mutant and its offspring. Thus, as far as fitness is concerned, the two mutants (and their offspring) can be considered equivalent. In our heuristics, the occurrence of a second (and also a third, fourth, ...) mutation within the given time window neither speeds up nor decelerates the (order of magnitude of) the time until the new fitness class is established in the population.

So $u(F)$ plays the role of a *refractory period*, in the sense that the fitness increments carried by contending mutations arriving within this period are lost. The probability that a given increment is *not* lost is determined via the expected waiting time for a (first) contending mutation to appear given the current value F , which is $v_1 := 1/(\mu\pi(F))$, and the expected duration $v_2 := u(F)$ of the refractory period. Specifically, by (4.15) and (4.20), the probability in question is

$$\frac{v_1}{v_1 + v_2} \sim \frac{1}{1 + C\mu \log N}. \quad (4.28)$$

Under this approximation, the expected per-day increase of the relative fitness, given its current value F , turns into

$$\mathbf{E}[\Delta F | F] \approx \frac{\mu\pi(F)\delta(F)}{1 + C\mu \log N} \sim \frac{\Gamma}{F^{2q+1}}, \quad (4.29)$$

where now

$$\Gamma = \frac{C\mu\varphi^2 \log \gamma}{1 + C\mu \log N}, \quad (4.30)$$

that is, the factor μ in (4.22) is replaced by $\mu/(1 + C\mu \log N)$. Now, taking the expectation over F in (4.29) yields

$$\mathbf{E}[\Delta F] \approx \Gamma \mathbf{E}\left[\frac{1}{F^{2q+1}}\right] \gtrsim \frac{\Gamma}{(\mathbf{E}[F])^{2q+1}}.$$

Here, the second step is due to Jensen's inequality.²⁶ Assuming a suitable concentration of the random variables in question around their expectations (which in theory would be justified by a dynamical law of large numbers result such as the one discussed in Section 4.2, and in practice is a crude way of moment closure) we arrive at the approximation

$$F_{[\tau/\Gamma]} \approx \mathbf{E}[F_{[\tau/\Gamma]}] \approx f(\tau) \text{ for large } N$$

with f as in (4.25). We may, therefore, approximate (as in Figure 4.4) the data by the function f , with the same values \hat{q} and $\hat{\Gamma}$ as before. The compound parameter Γ , however, has an internal structure different from the previous one (compare (4.30) with (4.22)). Solving (4.30) for μ gives

$$\mu = \frac{\Gamma}{C(\varphi^2 \log \gamma - \Gamma \log N)}; \quad (4.31)$$

for our current $\hat{\Gamma}$ and $\hat{\varphi}$, this yields $\hat{\mu} = 0.079$ and thus a better agreement between the simulated mean fitness and the approximating power law (and hence with the data), see Figure 4.5. Notably, $\hat{\mu}$ is of the same order of magnitude as $1/\log N = 0.15$; for an asymptotic analysis as $N \rightarrow \infty$, this would imply that the ratio (4.28) is bounded away from 0. For substantially higher mutation probabilities, the heuristics would break down

²⁶Note that $1/x^p$ is a convex function of x for any $p \geq 1$.

(cf. [38]) and a different asymptotic regime would apply (cf. [27]).

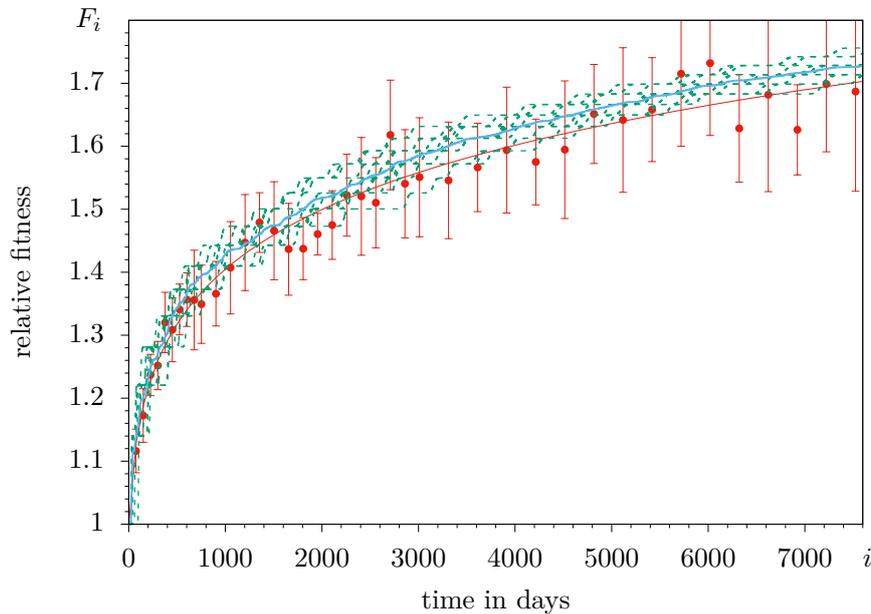


Figure 4.5.: Cannings simulation as in Figure 4.4, but with mutation probability $\hat{\mu} = 0.079$.

Let us now investigate the remaining discrepancy between the mean of the Cannings simulations and the approximating power law. Since the power law has been obtained via two approximations, namely the thinning heuristics and moment closure, it is interesting to quantify the contributions of these two sources of error. To this end, we simulate the evolution according to the heuristics rather than the Cannings model (see Algorithm 7). The result is shown in Figure 4.6. The simulation mean is very close to that of the Cannings simulation. We may conclude that the heuristics approximates the Cannings model very well, at least at the level of the mean values; the discrepancy between the Cannings simulation and the power law should therefore mainly be ascribed to moment closure. Note that the simulation of the heuristics yields smaller fluctuations than that of the Cannings model; this goes along with the fact that the model based on the heuristics contains fewer random elements than the Cannings model.

Let us finally comment on our choice $\hat{\varphi} = 0.14$. The denominator of (4.31) is strictly positive, and hence $\hat{\mu}$ is finite (and positive), as long as

$$\hat{\varphi} > \sqrt{\frac{\hat{\Gamma} \log N}{\log \gamma}} = 0.10.$$

The existence of such a lower bound on $\hat{\varphi}$ is plausible since the refractory period poses an upper bound to the rate of fixation events. Here we work with the value of $\hat{\varphi} = 0.14$ in order to stay reasonably far away from an undesirable ‘explosion’ of $\hat{\mu}$. With this choice, the mean number of fixed beneficial mutations in the simulations in Figure 4.6, averaged over the 12 runs, is 21; this is to be compared with the estimate of 60–110 fixed

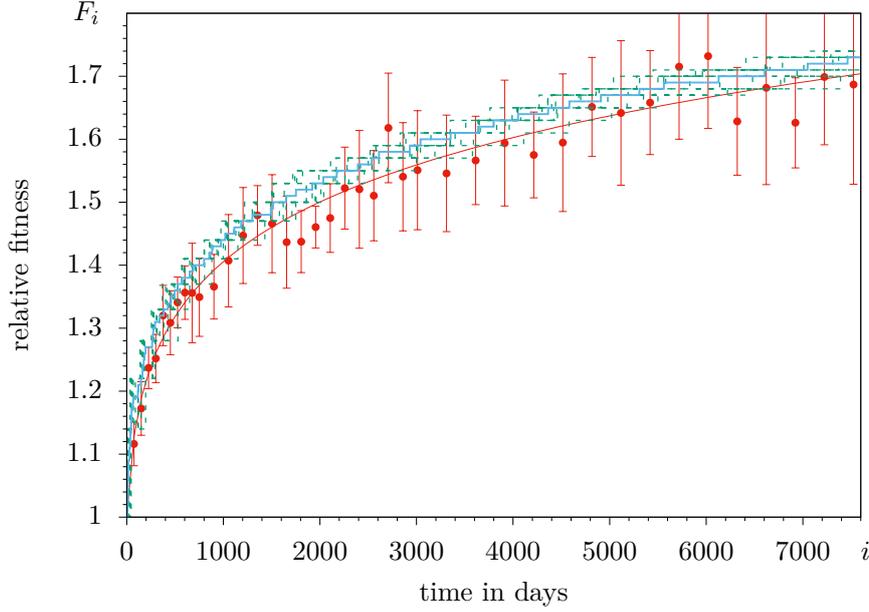


Figure 4.6.: Simulation using heuristics for deterministic increments. Parameters as in Figure 4.5. Mean number of clonal interference events: 24; mean number of established beneficial mutations: 21.

mutations observed in 50000 generations in [89], and of 100 fixed mutations observed in 60000 generations in [49], which both include neutral mutations.

Random beneficial effects. Let us now turn to random beneficial effects. To this end, we scale the fitness increments with a positive random variable X with density h and expectation $\mathbf{E}[X] = 1$. We assume throughout that $\mathbf{E}[X^2] < \infty$ to ensure that all quantities required in what follows are well-defined. Taking into account the dependence on X , the quantities in (4.11)–(4.13), (4.15) and (4.20) turn into

$$\delta(F, X) = X \frac{\varphi}{Fq}, \quad (4.32a)$$

$$\sigma(F) = \frac{\log \gamma}{F} \text{ (as before),} \quad (4.32b)$$

$$s(F, X) = \delta(F, X) \sigma(F), \quad (4.32c)$$

$$\pi(F, X) \approx C s(F, X), \quad (4.32d)$$

$$u(F, X) = \frac{\log N}{s(F, X)}. \quad (4.32e)$$

Note first that large X implies large s and hence small u and vice versa; and second that (4.32d) is an approximation, whereas in (4.15) we have asymptotic equivalence. The following Poisson picture will be central to our heuristics: the process of *beneficial mutations* with scaled effect x that arrives at time τ has intensity $\mu d\tau h(x) dx$ with points $(\tau, x) \in \mathbb{R}_+ \times \mathbb{R}_+$. And in fitness background $\approx F$, we denote by Π the Poisson process of

contending mutations, i.e. those beneficial mutations that survive drift (but not necessarily go to fixation), which has intensity $\mu d\tau h(x)\pi(F, x) dx$ on $\mathbb{R}_+ \times \mathbb{R}_+$.

We now develop a refined version of the *Gerrish-Lenski heuristics for clonal interference* and adapt it to the context of our model. If, in the fitness background $\approx F$, two contending mutations (τ, x) and (τ', x') appear at $\tau < \tau' < \tau + u(F, x)$, then the first one outcompetes ('kills') the second one if $x' \leq x$, and the second one kills the first one if $x' > x$. Thus, neglecting interactions of higher order, given that a contending mutation arrives at (τ, x) in the fitness background $\approx F$, the probability that it does not encounter a killer in its past is

$$\overleftarrow{\chi}(F, x) := \exp\left(-\int_x^\infty \mu \pi(F, y) u(F, y) h(y) dy\right), \quad (4.33)$$

whereas the probability that it does not encounter a killer in its future is

$$\overrightarrow{\chi}(F, x) := \exp\left(-u(F, x) \int_x^\infty \mu \pi(F, x') h(x') dx'\right). \quad (4.34)$$

Using (4.32), $\overleftarrow{\chi}(F, x)$ is approximated by

$$\overleftarrow{\psi}(x) := \exp\left(-\mu C \log N \int_x^\infty h(y) dy\right),$$

whereas $\overrightarrow{\chi}(F, x)$ is approximated by

$$\overrightarrow{\psi}(x) := \exp\left(-\mu \frac{C \log N}{x} \int_x^\infty x' h(x') dx'\right). \quad (4.35)$$

Notably, neither $\overleftarrow{\psi}$ nor $\overrightarrow{\psi}$ depend on F . Thus, setting $\overleftarrow{\chi} := \overleftarrow{\chi} \overrightarrow{\chi}$ and analogously $\overleftrightarrow{\psi} := \overleftarrow{\psi} \overrightarrow{\psi}$, we obtain, as an analogue of (4.29), the expected (per-day) increase of F , given the current value of F , as

$$\begin{aligned} \mathbf{E}[\Delta F | F] &\approx \mu \int_0^\infty \delta(F, x) \pi(F, x) \overleftarrow{\chi} \overrightarrow{\chi}(F, x) h(x) dx \\ &\approx \frac{C \mu \varphi^2 \log \gamma}{F^{2q+1}} \int_0^\infty x^2 \overleftrightarrow{\psi}(x) h(x) dx = \frac{\Gamma}{F^{2q+1}}, \end{aligned} \quad (4.36)$$

where

$$\Gamma := C \mu \varphi^2 \log(\gamma) I(\mu) \quad (4.37)$$

and $I(\mu) := \mathbf{E}\left[\overleftrightarrow{\psi}(X) X^2\right]$ is the integral in (4.36) whose parameter μ still has to be determined. Similarly as in the previous paragraph on deterministic beneficial effects, the assumption of a suitable concentration of the random variable ΔF around its conditional expectation allows us to take (4.36) into

$$F_{[\tau/\Gamma]} \approx \mathbf{E}[F_{[\tau/\Gamma]}] \approx f(\tau)$$

with f as in (4.25) and we will refer to this approximation step as ‘moment closure’. The composite parameter Γ can be estimated from the empirical data in the same way as described at the end of Section 4.2.

In order to estimate μ and φ with the help of the observed mean fitness increment of the first fixed beneficial mutation (in analogy with (4.27)), we derive, in our Poisson model, the *expectation of the scaled effect of the first among the contending mutations* (in fitness background $F = 1$) *that is not killed*. To this end, we consider a sequence of points $(\mathcal{T}_j, X_j)_{j \geq 1}$ in Π (the Poisson process of contending mutations) that is strictly monotonic increasing in both coordinates and inductively defined as follows. (\mathcal{T}_1, X_1) is the point in Π with the smallest τ -coordinate, and given (\mathcal{T}_j, X_j) , $(\mathcal{T}_{j+1}, X_{j+1})$ is the point in Π with

$$\mathcal{T}_{j+1} = \min\{\tau : (\tau, x) \in \Pi, \tau > \mathcal{T}_j, x > X_j\}.$$

Again we say that $(\mathcal{T}_{j+1}, X_{j+1})$ kills (\mathcal{T}_j, X_j) , if $\mathcal{T}_{j+1} < \mathcal{T}_j + u(X_j)$.²⁷ Let

$$Z := \min\{j \geq 1 : \mathcal{T}_{j+1} > \mathcal{T}_j + u(X_j)\},$$

i.e. (\mathcal{T}_Z, X_Z) is the earliest among the points (\mathcal{T}_j, X_j) , $j = 1, 2, \dots$, that is *not killed*. The point (\mathcal{T}_Z, X_Z) is thus called the (first) *winner*; at time \mathcal{T}_Z , the relative fitness of the population jumps from 1 to $1 + \varphi X_Z$.

Our aim is to find the distribution of the x -coordinate of the winner,

$$\mathbf{P}[X_Z \in dx], \quad x \geq 0. \quad (4.38)$$

From elementary properties of Poisson processes we infer that, given (\mathcal{T}_j, X_j) , the waiting time $W_{j+1} := \mathcal{T}_{j+1} - \mathcal{T}_j$ is exponentially distributed with parameter $\mu \int_{X_j}^{\infty} \pi(y) h(y) dy$.

Hence

$$\mathbf{P}[W_{j+1} > u(x) \mid X_j = x] = \chi(x),$$

with $\chi(x) := \vec{\chi}(1, x)$ from (4.34). Moreover, given (\mathcal{T}_j, X_j) , the random variables W_{j+1} and X_{j+1} are independent, and X_{j+1} has the conditional density

$$\mathbf{P}[X_{j+1} \in dx \mid X_j = y] = \frac{\pi(x) h(x) dx}{\int_y^{\infty} \pi(y') h(y') dy'} =: \rho(x \mid y) dx \quad (4.39)$$

for $x \geq y \geq 0$. Consequently, the conditional probability that the j -th of the increasing points is the winner, given that all the previous ones have been killed, is

$$\mathbf{P}[Z = j \mid Z \geq j, X_j = y] = \chi(y),$$

²⁷As long as we assume $F = 1$, we suppress the first argument in the functions defined in (4.32) for notational convenience.

whereas the conditional probability to proceed and see the next killer at dx is

$$\mathbf{P}[Z \geq j + 1, X_{j+1} \in dx \mid Z \geq j, X_j = y] = (1 - \chi(y)) \rho(x \mid y) dx.$$

With $x_0 := 0$, this gives the formula for the joint distribution of X_1, \dots, X_Z and Z :

$$\begin{aligned} \mathbf{P}[X_1 \in dx_1, \dots, X_{\ell-1} \in dx_{\ell-1}, X_\ell \in dx, Z = \ell] \\ = \prod_{j=1}^{\ell-1} \rho(x_j \mid x_{j-1}) (1 - \chi(x_j)) dx_j \rho(x \mid x_{\ell-1}) \chi(x) dx. \end{aligned} \quad (4.40)$$

The density $\mathbf{P}[X_Z \in dx, Z = \ell]$ arises by integrating (4.40) over $x_1, \dots, x_{\ell-1}$, under the constraints $0 \leq x_1 \leq \dots \leq x_{\ell-1} \leq x$. Using (4.39), we see from (4.40) that

$$\mathbf{P}[X_Z \in dx, Z \leq \ell] = \sum_{k=1}^{\ell} \mathbf{P}[X_Z \in dx, Z = k] = \pi(x) \chi_\ell(x) h(x) dx. \quad (4.41)$$

Here χ_ℓ may be read off the product formula (4.40), plays the role of an additional reweighting factor, and coincides with (4.34) in the case $\ell = 1$. Then, for the density of X_Z conditional on $Z \leq \ell$, we obtain

$$\mathbf{P}[X_Z \in dx \mid Z \leq \ell] = \frac{\pi(x) \chi_\ell(x) h(x) dx}{\int_0^\infty \pi(x) \chi_\ell(x) h(x) dx}, \quad (4.42)$$

which should be very close to (4.38) for ℓ not too small.

Consequently, with the approximations (4.32) as well as (4.35), and ψ_ℓ taking the place of χ_ℓ in the approximate analogue of (4.41),

$$\mathbf{E}[\delta(X_Z) \mid Z \leq \ell] = \varphi \mathbf{E}[X_Z \mid Z \leq \ell] \approx \varphi \frac{\mathbf{E}[\psi_\ell(X) X^2]}{\mathbf{E}[\psi_\ell(X) X]} =: \varphi \zeta_\ell(\mu). \quad (4.43)$$

Note that, under the assumptions on X , $\zeta_\ell(\mu)$ (as an approximation of the expectation of the first scaled beneficial effect that goes to fixation) is parametrised by μ (via ψ_ℓ) and well defined for any μ , since $0 < \psi_\ell \leq 1$.

Then, again for ℓ not too small, the left-hand side of (4.43) may be a good approximation for the observed value of the mean fitness increment due to the first fixed beneficial mutation, which we denote by \mathfrak{d}_1 . Indeed, (4.37) together with (4.43) renders the system of equations

$$\frac{\mu I(\mu)}{(\zeta_\ell(\mu))^2} = \frac{\Gamma}{C \mathfrak{d}_1^2 \log \gamma}, \quad (4.44a)$$

$$\varphi = \frac{\mathfrak{d}_1}{\zeta_\ell(\mu)} \quad (4.44b)$$

where μ , as the solution of (4.44a), determines φ via (4.44b). We will carry out this program with $\ell = 3$ for two special choices of h in the remainder of this section. Let us anticipate that numerical evaluations show that the left-hand side of (4.44a) is monotone increasing in μ on $[0, 1]$ for both choices. For (4.44a) to have a solution, it is therefore required that

$$\mathfrak{d}_1 > \sqrt{\frac{\Gamma}{C \log(\gamma) I(1)}} \zeta_3(1), \quad (4.44c)$$

which will serve as a lower bound for the estimate $\widehat{\mathfrak{d}}_1$.

The analysis so far allows to conclude that, as long as the above described approximation may be relied on, the *mean fitness curve* observed in [97] *can be described by any distribution of fitness effects*, provided the mutation probability is chosen according to (4.44a) (and provided that (4.44a) has a solution). In particular, the *epistasis parameter* q is *not affected by the distribution* of X .

Exponentially distributed beneficial effects. For definiteness, we now turn to random beneficial effects where X follows $\text{Exp}(1)$, the exponential distribution with parameter 1. This is the canonical choice since strongly-beneficial mutations appear to be exponentially distributed; the experimental evidence is reviewed in [33], and it confirms theoretical predictions (see [45], [75]). The distribution of slightly-beneficial mutations is less well known, but these mutations contribute little to the adaptive process. Thus, by (4.44), we have $\widehat{\mu} = 0.11$ and hence $\widehat{\varphi} = 0.069$ for $\widehat{\mathfrak{d}}_1 = 0.20 \gtrsim 0.16$ for this choice of the distribution of X . Figure 4.7 shows the corresponding Cannings simulations, and Figure 4.8 displays the simulations according to the heuristics. The agreement of the simulation mean with the approximating power law is now nearly perfect. The fluctuations, however, are smaller in the simulations than in the experiment. As argued in Section 4.2 in the context of the first reality check, this may be explained by the constant parameters assumed by the model, whereas parameters do vary across replicate populations in the experiment.

Let us also mention the degree of polymorphism observed in the Cannings simulations of Figure 4.7. Counting a type as ‘present’ if its frequency is at least 20%, it turns out that, on average, the population is monomorphic on 89.1% of the days; it contains two types on 10.6% of the days, and three types on 0.3% of the days. Thus, in the finite system, some polymorphism is present, but it is not abundant. Recall that our model does not consider neutral mutations, and thus the low level of (fitness) polymorphism observed in the simulations does not contradict the high level of genetic diversity observed in experiments (see [89]).

Beneficial effects with a Pareto distribution. As argued already, the exponential distribution seems to be the most realistic choice for beneficial mutation effects. The theory developed above, however, holds for arbitrary probability distributions on the positive

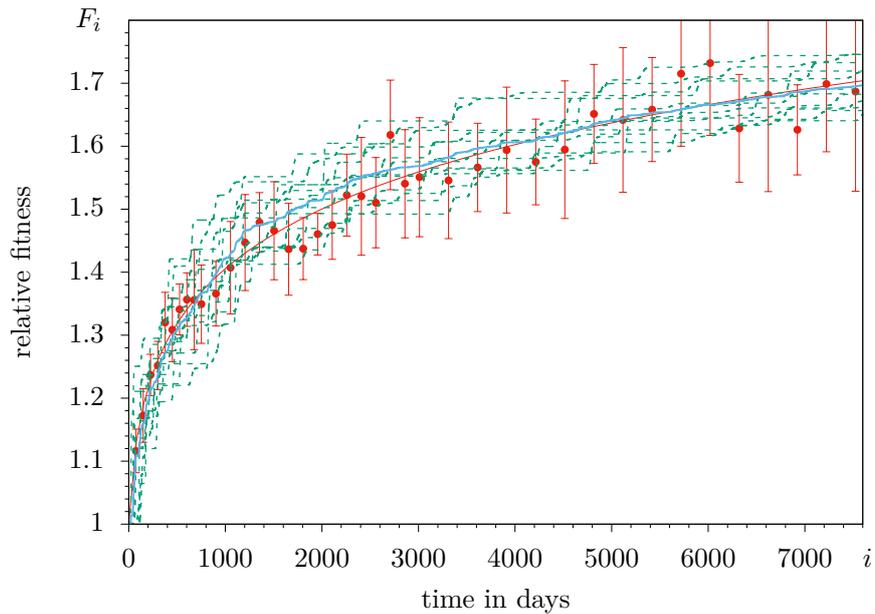


Figure 4.7.: Simulations of the Cannings model with X following $\text{Exp}(1)$ and parameters $\hat{\varphi} = 0.069$, and mutation probability $\hat{\mu} = 0.11$.

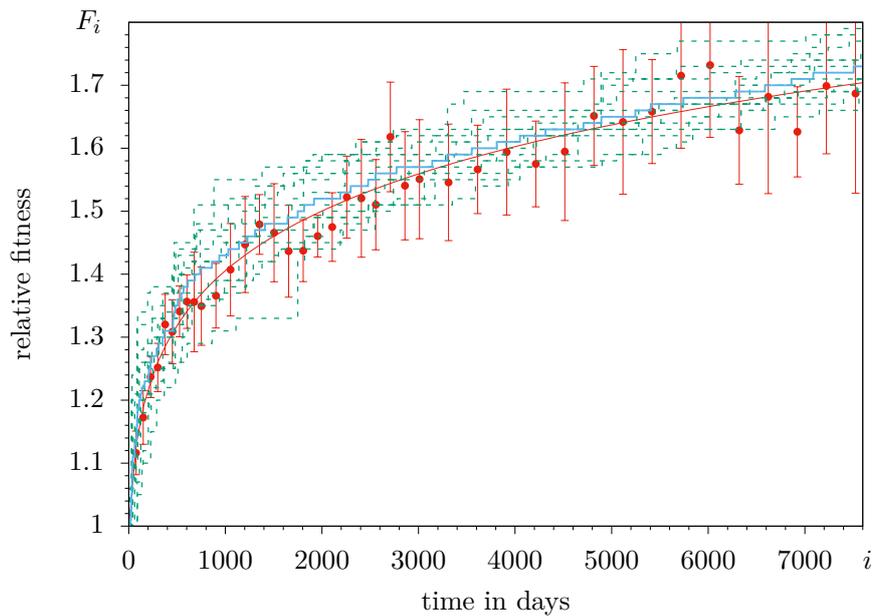


Figure 4.8.: Simulations using Gerrish-Lenski heuristics with X following $\text{Exp}(1)$ and parameters as in Figure 4.7. Mean number of clonal interference events with $x' \leq x$: 7.9; mean number of clonal interference events with $x' > x$: 7.8; mean number of established beneficial mutations: 23.

half axis that have expectation 1 and a finite second moment. Furthermore, the analysis of the heuristics indicates that the results are, in fact, independent of the distribution, provided the compound parameter Γ is interpreted in the appropriate way. It is therefore

interesting to explore whether this conclusion may be verified by simulations. In order to push our conjecture to the limits, we choose X distributed according to a (*shifted*) *Pareto distribution* (see [35, Ch. II.4] or [88, Ex. 2.19]) with shape parameter λ as given by the density

$$h(x) = \begin{cases} 0, & x < 0 \\ \frac{\lambda}{\lambda-1} \left(\frac{\lambda-1}{x+(\lambda-1)} \right)^{\lambda+1}, & x \geq 0. \end{cases} \quad (4.45)$$

The parameter $\lambda \geq 0$ controls which of the moments of X exist. For $0 < \lambda \leq 1$, the expectation is infinite; for $1 < \lambda < 2$, the expectation is 1 but the second moment is infinite; for $2 < \lambda \leq 3$, $\mathbf{E}[X] = 1$ and $\mathbf{E}[X^2] = 2(\lambda - 1)/(\lambda - 2)$ but the third moment is infinite; and similarly for larger λ . We work with $\lambda = 2.5$ here; this implies that there is no restriction in applying our analysis. Proceeding in analogy with the case of exponentially distributed beneficial effects, we simulate both the Cannings model and the heuristics and compare them with the approximating power law. The result is shown in Figure 4.9 and Figure 4.10.

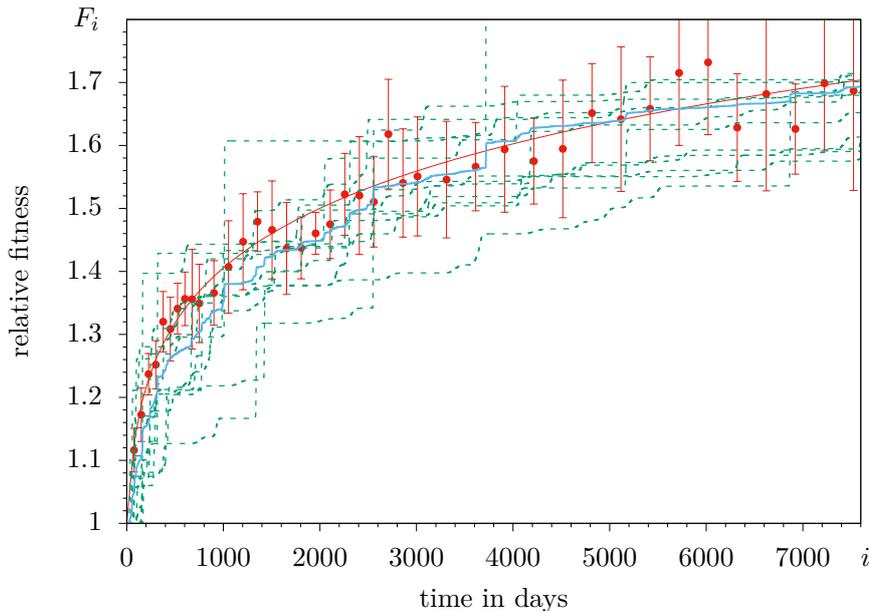


Figure 4.9.: Simulations of the Cannings model with X following the (shifted) Pareto distribution with density h of (4.45). Parameters: $\lambda = 2.5$, $\hat{\varphi} = 0.020$, and $\hat{\mu} = 0.37$.

As was to be expected, the mean is still well described by the approximating power law, but the fluctuations are enhanced relative to the case of the exponential distribution (note that now $\mathbf{E}[X^2] = 6$ in contrast to $\mathbf{E}[X^2] = 2$ in the case of $\text{Exp}(1)$, and thus, by (4.44), we have $\hat{\mu} = 0.37$ and hence $\hat{\varphi} = 0.020$ for $\hat{\mathfrak{d}}_1 = 0.12 \gtrsim 0.10$). Compared to the experiment, the fluctuations are unrealistically large; an effect distribution with high variance therefore does not appear to be close to the truth.

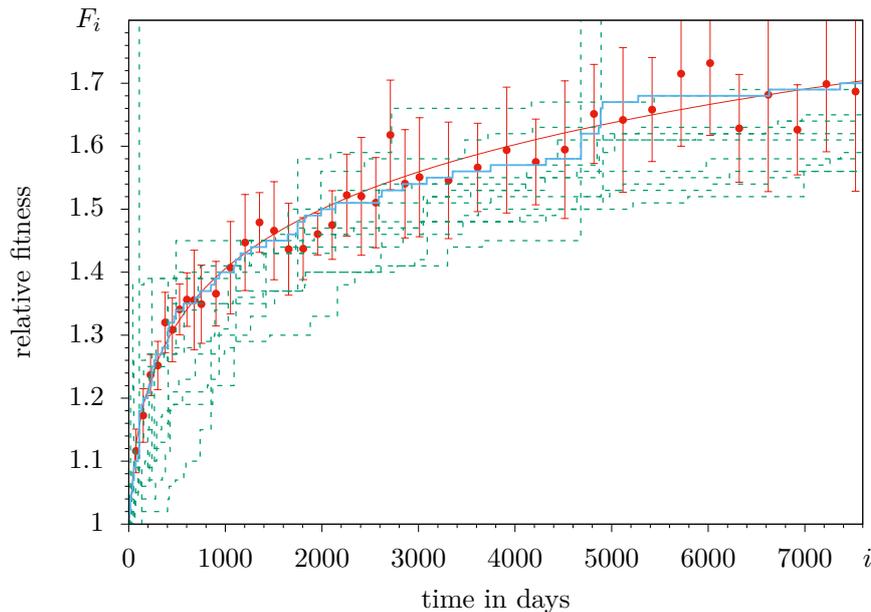


Figure 4.10.: Simulations using Gerrish-Lenski heuristics with X following the (shifted) Pareto distribution and parameters as in Figure 4.9. Mean number of clonal interference events with $x' \leq x$: 13; mean number of clonal interference events with $x' > x$: 9.1; mean number of established beneficial mutations: 21.

4.4. Simulation algorithms

Let us briefly describe the two algorithms we have used to simulate our model. Before we come to the details, let us say a few words about notation and strategy. We will throughout use the framework (4.32), which reduces to (4.11)–(4.13), (4.15) and (4.20) in the case of deterministic beneficial effects, where $X \equiv 1$, that is, the distribution of X is a point measure on 1. (But note that, in paragraph about Random beneficial effects, we assume that X has a density; this implies that any two realisations of X are different with probability 1, so that there is a clear ‘winner’ in the Gerrish-Lenski heuristics. The analysis thereof therefore does not carry over to the deterministic case.) *Curly* symbols indicate *sets* of values, whereas *bold* symbols indicate *lists* and $\bullet^{(k)}$ their k -th element. By slight abuse of notation, we denote by $\delta \left(\overleftarrow{\delta} \right)$ the increment of relative fitness (4.32a) for the current (previous) mutation.

The Cannings model. Algorithm 6 performs an individual-based simulation of the Cannings model with selection, as formulated in Section 4.2. Its iterations are based on real-world days i . The algorithm keeps track of the sizes \mathcal{N}_j of the classes (or subpopulations) of individuals that have reproduction rate \mathcal{R}_j , $j \geq 1$. As long as n_{typ} , the number of different reproduction rates in the population, equals 1, the population is homogeneous, so that the intraday growth and subsequent sampling do not change the current state. If $n_{\text{typ}} > 1$, we use the fact that the clone size at time σ in a Yule process with branching

rate \mathcal{R}_j started by a single individual is 1 plus a random variable that follows $\text{Geo}(e^{-\mathcal{R}_j\sigma})$, the geometric distribution²⁸ with parameter $e^{-\mathcal{R}_j\sigma}$ (cf. [34, Ch. XVII.3] or [26, Ch. 1.3.3]). The size of the corresponding subpopulation at time σ is then \mathcal{N}_j plus the sum of \mathcal{N}_j independent copies of the geometric random variable. This sum follows $\text{NB}(\mathcal{N}_j, e^{-\mathcal{R}_j\sigma})$, the negative binomial distribution with parameters \mathcal{N}_j and $e^{-\mathcal{R}_j\sigma}$, cf. [34, Ch. VI.8] or [88, pp. 168/169]. The only point where each individual must be treated separately is the sampling step, where $N = 5 \cdot 10^6$ new founder individuals are drawn without replacement from the $\approx 5 \cdot 10^8$ descendants. After the sampling, the number of mutation events is drawn from $\text{Poi}(\hat{\mu})$, the Poisson distribution with parameter $\hat{\mu}$ (line 13). The affected individuals are then chosen *uniformly without replacement* from among the N new founders.

Algorithm 6: Simulating Lenski's experiment (Cannings model)

Input: User chosen density law of X and parameters ι_{\max} , \hat{q} , $\hat{\mu}$, $\hat{\varphi}$.

```

1 Initialise  $k = 0$ ,  $\sigma = 1$ ,  $n_{\text{typ}} = 1$ ,  $n_{\text{mut}} = 0$ ,  $\mathcal{R} = \{1\}$ ,  $\mathcal{N} = \{N\}$ .
2 while  $k < \iota_{\max}$  do
    // Length of intraday growth time
3   Solve (4.6), i.e.  $\sum_{j=1}^{n_{\text{typ}}} \mathcal{N}_j e^{\mathcal{R}_j\sigma} = \gamma N$ , to obtain  $\sigma$ .
4   Set  $F^{(k)}$  according to (4.8).
5   if  $n_{\text{typ}} > 1$  then
        // Intraday population growth
6        $n_{\text{des}} \leftarrow 0$ .
7       for  $j = 1, \dots, n_{\text{typ}}$  do
8         Draw  $D \sim \text{NB}(\mathcal{N}_j, e^{-\mathcal{R}_j\sigma})$  and set  $n_{\text{des}} \leftarrow n_{\text{des}} + \mathcal{N}_j + D$ .
        // Interday sampling
9       Draw sample  $\{j_1, \dots, j_N\}$  without replacement from  $\{1, \dots, n_{\text{des}}\}$  and set
         $\mathcal{N} = \{\mathcal{N}_1, \dots, \mathcal{N}_{n_{\text{typ}}}\}$  accordingly.
10      for  $j = 1, \dots, n_{\text{typ}}$  do
11        if  $\mathcal{N}_j = 0$  then
12          Remove type  $j$  and set  $n_{\text{typ}} \leftarrow n_{\text{typ}} - 1$ .
        // Mutation
13      Draw  $n_{\text{mut}} \sim \text{Poi}(\hat{\mu})$  and set  $n_{\text{typ}} \leftarrow n_{\text{typ}} + n_{\text{mut}}$ .
14      if  $n_{\text{mut}} > 0$  then
15        Draw sample  $\{i_1, \dots, i_{n_{\text{mut}}}\}$  without replacement from  $\{1, \dots, N\}$ .
16        for  $j = 1, \dots, n_{\text{mut}}$  do
17           $\mathcal{N}_{i_j} \leftarrow \mathcal{N}_{i_j} - 1$  and  $\mathcal{N} \leftarrow \mathcal{N} \cup \{1\}$ .
18          Draw  $X$  and set  $\mathcal{R} \leftarrow \mathcal{R} \cup \{\mathcal{R}_{i_j} + \delta(\mathcal{R}_{i_j}, X)\}$  acc. to (4.32a).
19          if  $\mathcal{N}_{i_j} = 0$  then
20            Remove type  $i_j$  and set  $n_{\text{typ}} \leftarrow n_{\text{typ}} - 1$ .
21       $k \leftarrow k + 1$ .
22 return  $F$ .
```

²⁸ $\text{Geo}(p)$ is the distribution of the number of failed coin tosses before the first success (success prob. p).

The thinning heuristics. Algorithm 7 unifies the two versions of the thinning heuristics of Section 4.3. We now only keep track of substitutions that effectively lead to an increase of the relative fitness, and thus have a homogeneous population in every iteration k . The number k counts the fixation events, and the vector $\boldsymbol{\iota}$ holds the times at which they occur. More precisely, mutations appear after waiting times Δ_ι following $\text{Exp}(\hat{\mu})$ (approximating the discrete $\text{Geo}(\hat{\mu})$ -distribution). For every such mutation, it is decided whether or not it survives drift by drawing a Bernoulli random variable with success probability π according to (4.32d) (line 13). If the mutation survives, it is queried whether it survives clonal interference. We simulate this by first adding the increment $\overleftarrow{\delta}$ due to a ‘first’ mutation to the mean fitness, and then adding the additional increment $\delta - \overleftarrow{\delta}$ due to the ‘second’ mutation if it outcompetes the former. For the choice $X \equiv 1$, this means that the first out of two competing mutations wins; the case of **A fitter mutation appeared** in line 7 can never occur for deterministic increments.

Algorithm 7: Simulating Lenski's experiment (thinning heuristics)

Input: User chosen law of X and parameters $\iota_{\max}, \hat{q}, \hat{\mu}, \hat{\varphi}$.

```

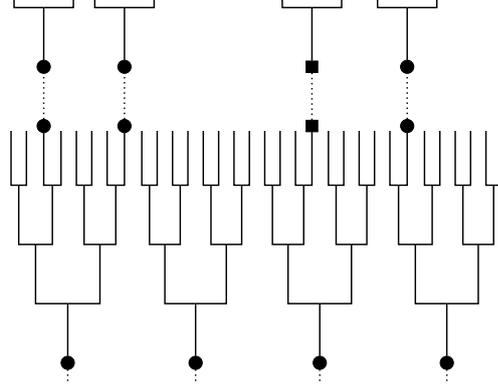
1 Initialise  $k = 0, \iota = 0, \delta = \overleftarrow{\delta} = 0, \boldsymbol{\iota}^{(0)} = 0, \mathbf{F}^{(0)} = 1$ .
2 while not terminated, i.e.  $\boldsymbol{\iota}^{(k)} + \iota \leq \iota_{\max} \wedge k \leq k_{\max}$  do
    // Not within refractory period according to (4.32e), (4.32c), (4.32b)
3   if  $\iota > \log(N)/(\overleftarrow{\delta} \sigma(\mathbf{F}^{(k)}))$  then
    // Beneficial mutant becomes fixed unrivalled
4      $(\boldsymbol{\iota}^{(k+1)}, \mathbf{F}^{(k+1)}) \leftarrow (\boldsymbol{\iota}^{(k)} + \iota, \mathbf{F}^{(k)} + \delta)$ .
5      $(\iota, k, \overleftarrow{\delta}) \leftarrow (0, k + 1, \delta)$ .
6   else
    // A fitter mutation appeared
7     if  $\delta > \overleftarrow{\delta}$  then
8        $(\boldsymbol{\iota}^{(k+1)}, \mathbf{F}^{(k+1)}) \leftarrow (\boldsymbol{\iota}^{(k)} + \iota, \mathbf{F}^{(k)} + \delta - \overleftarrow{\delta})$ .
9        $(\iota, k, \overleftarrow{\delta}) \leftarrow (0, k + 1, \delta)$ .
    // Occurrence of a next mutant to become dominant
10  do
11    Draw  $X$  and set  $\delta \leftarrow \hat{\varphi} X (\mathbf{F}^{(k)})^{-\hat{q}}$  according to (4.32a).
12    Draw  $\Delta_\iota$  following  $\text{Exp}(\hat{\mu})$  and set  $\iota \leftarrow \iota + \Delta_\iota$ .
13  while  $S$  following  $\text{Ber}(C \delta \sigma(\mathbf{F}^{(k)}))$  is unsuccessful according to (4.32d)
14 return  $(\boldsymbol{\iota}, \mathbf{F})$ .
```

4.5. Discussion

We have, so far, postponed a detailed comparison with the model and the results in [97]. We now have everything at hand to do so.

Modelling aspects. Both the WRL model and ours lead to power laws, (4.2) and (4.25), which are of the same form. But the modelling assumptions differ in relevant details, with

Figure 4.11: Synchronous growth model as used in [42], with equally-sized clones at the end of the day (here, $\gamma = 8$); compare Figure 4.1.



consequences for the interpretation of the parameters. Here and below we use a tilde to distinguish the quantities belonging to the WRL model from our corresponding quantities. The main difference is that in [97] the experiment is described with a discrete generation scheme given by $\log_2 \gamma$ (≈ 6.6) doublings during one daily growth phase, see Figure 4.11. This neglects the variability that comes from a continuous-time intraday reproduction mechanism, and affects the WRL analogue to our formula (4.15) for the probability to survive drift. The latter is stated in (S1) of their Supplementary Text, reads

$$\tilde{\pi} = \tilde{\pi}(\tilde{s}) = 4\tilde{s}, \quad (4.46)$$

and relies on [42], Appendix 1. In line with the generation scheme of Figure 4.11, \tilde{s} is the selective advantage in each of the $\log_2 \gamma$ generations per day. At the end of the day, the population has increased from size N to size γN and consists of N clones, each of size γ . A sampling of N individuals without replacement thus leads to a pair coalescence probability of $(\gamma - 1)/(\gamma N)$, and hence to an offspring variance per day of

$$\tilde{v} \sim \frac{\gamma - 1}{\gamma}; \quad (4.47)$$

note the factor of 2 between \tilde{v} and our v in (4.16), which comes from a size-biasing effect due to the sampling from clones of random size. Since \tilde{s} is related to one ‘doubling generation’, the selective advantage *per day* is

$$\tilde{s}_d \approx \tilde{s} \log_2 \gamma. \quad (4.48)$$

Now, Haldane’s formula (4.17) related to the daily rhythm gives

$$\tilde{\pi} \approx \frac{\tilde{s}_d}{\tilde{v}_d/2},$$

and, with (4.46), this yields a per-day offspring variance $\tilde{v}_d \approx \log_2 \gamma$, which is significantly different from \tilde{v} in (4.47) for $\gamma = 100$. Thus, we see that the ansatz in [97] combined with [42] leads to an ambiguously defined offspring variance per day.

Moreover, at the end of the Materials and Methods section in the Supplement of [97] the difference between the new and the old relative fitness to the (per generation) selective advantage of a mutant is related as follows:

$$w_{\text{new}} = w(1 + \tilde{s}) \quad (4.49)$$

with \tilde{s} from (4.46). Here

$$w = w_i = \frac{\log \tilde{a}}{\log \tilde{b}}, \quad (4.50)$$

with the growth factors $\tilde{a} = Y_i(T_i)/Y_i(0)$ and $\tilde{b} = Y_0(T_i)/Y_0(0)$ as in (4.1). They are not explicit about an intraday growth model, so one should think of $Y_i(0)$, $Y_0(0)$, $Y_i(T_i)$ and $Y_0(T_i)$ as the number of individuals at the beginning and the end of the competition experiment. For a consistent definition of the selective advantage per day, it is inevitable to use the growth factors a_{new} and a related to one day; then, according to (4.14), one has

$$s_d = \frac{a_{\text{new}} - a}{a} \sim \log \frac{a_{\text{new}}}{a}. \quad (4.51)$$

In principle, a may (and will) differ from the \tilde{a} in the definition of w . At least in a model with intraday exponential growth, however, the definition of w in (4.50) becomes independent of T_i ; we may (and will) therefore use the growth factors $a = Y_i(\sigma_i)/Y_i(0)$ and $b = Y_0(\sigma_i)/Y_0(0)$ instead of \tilde{a} and \tilde{b} in (4.50). Then (4.50) implies

$$\frac{w_{\text{new}}}{w} = \frac{1}{\log a} \left(\log \left(\frac{a_{\text{new}}}{a} \right) + \log a \right), \quad (4.52)$$

which by (4.51) yields

$$w_{\text{new}} = w \left(1 + \frac{s_d}{\log a} \right), \quad (4.53)$$

or equivalently, using (4.50) again,

$$w_{\text{new}} - w = \frac{s_d}{\log b}. \quad (4.54)$$

Under the assumption of an intraday exponential growth we have (as long as the populations are nearly homogeneous):

$$a \approx e^{r\sigma}, \quad b \approx e^\sigma, \quad w \approx r, \quad r\sigma \approx \log \gamma. \quad (4.55)$$

Thus (4.54) translates into

$$s_d \approx \frac{1}{r} (r_{\text{new}} - r) \log \gamma, \quad (4.56)$$

which also results from combining (4.12) and (4.13) and equating F and r . This shows that the runtime effect discussed in Section 4.2 is already implicit in the definition (4.50) of w as the ratio of logarithms of growth factors, as soon as one uses a model with intraday

exponential growth. Let us emphasise again that this runtime effect is a consequence of the design of Lenski's experiment; it would be absent in a variant of the experiment in which sampling occurs at a given fixed time before the onset of the starvation phase. Furthermore, comparing (4.53) with (4.49) and using (4.55) gives

$$s_d = \tilde{s} \log a \approx \tilde{s} \log \gamma.$$

Comparing with (4.48), this shows that

$$s_d = \frac{\log \gamma}{\log_2 \gamma} \tilde{s}_d,$$

which points to a certain inconsistency inherent in \tilde{s}_d .

Another issue worth to compare is the interpretation of *diminishing returns epistasis*, and the corresponding translation between the exponent g in the WRL model and the exponent q in ours. Formula (S1) in [97] says that the multiplicative effect on r has expected size $1/\alpha$; this corresponds to an additive effect on r of expected size $\delta := r/\alpha$. Thus, the ansatz (4.11) translates into

$$\frac{1}{\alpha} = \frac{\varphi}{r^{q+1}}.$$

On the other hand, formula (S9) in [97] says that

$$\alpha = c e^{g \log r},$$

which implies that $g = q + 1$. The choice $g = 1$ in the WRL model (or equivalently, $q = 0$ in ours, cf. (4.2) and (4.25)) corresponds to *additive* increments on the Malthusian fitness that do not depend on the current value of the latter, see (4.11). It is this case of constant additive increments which may be appropriately addressed as the *absence of epistasis*. More precisely, in *continuous time* (as considered here for the intraday dynamics), additive fitness increments correspond to independent action of mutations and hence to absence of epistasis (cf. [36], [21, pp. 48 and 74]); in *discrete time*, the same would be true of multiplicative increments. Consequently, $q = g - 1$ can be seen as an exponent describing the effect of epistasis. With this interpretation, a (slight) concavity of the mean fitness curve is caused by the runtime effect (and hence by the design of the experiment) even in the absence of epistasis. This fact, which is due to the runtime effect, is sometimes overlooked when interpreting the mean fitness curve; see, for example, [48].

A substantial part of the derivations in [97] deals with incorporating the Gerrish-Lenski heuristics for *clonal interference* into their model. The fact that they work with multiplicative fitness increments and various approximations complicates the translation between the time-scaling constant in their power law (S16) (that we subsume as β in (4.2)) and our time-scaling constant Γ (see (4.25) and (4.37)). We refrain from pursuing the details here; but let us emphasise that (4.32) together with the calibrations discussed in the

surrounding paragraph applies to arbitrary random (additive) fitness effects with finite second moments.

Analytic and simulation results. We have presented three lines of results. First, rigorous results for the relative mean fitness in terms of a law of large numbers in the limit $N \rightarrow \infty$ for deterministic beneficial effects in a regime of weak mutation and moderately strong selection. Second, we have derived transparent analytic expressions for the *expected* mean fitness in a finite- N system by means of heuristics of Gerrish-Lenski type and a moment closure approximation (which is also used in [97]). The beneficial effects may be either deterministic (and then require a specific thinning heuristics), or random with an arbitrary density. In the latter case we have developed a refinement of the original Gerrish-Lenski heuristics. Briefly stated, this refinement does not only consider the thinning factor (4.34) coming from *future* interfering mutations, but also the thinning factor (4.33) coming from *past* ones. This makes the heuristics consistent with its verbal description, which says that ‘if two contending mutations appear within the time required to become dominant in the population, then the fitter one wins’. A refinement that also includes thinning due to past competitors was suggested in [41], but contains less detail; in particular, it does not allow conclusions about the distribution of the effects of the ‘winning’ mutations.

For reasons of calibration, we have indeed established an approximate analytic expression (4.43) for the expected scaled effect of the first beneficial mutation that goes to fixation. This introduces a *size bias* into the distribution of beneficial effects (see (4.42)), similar to the descriptions in [84] and [97] in the case of the exponential distribution.

As it turned out, the analytic expressions are *robust*. In particular, the estimate of q is neither affected by clonal interference nor by the choice of the distribution. What changes is the internal structure of the compound parameter Γ , but for any given estimate $\hat{\Gamma}$, the mutation probability and scaling of beneficial effects may be arranged appropriately (provided X has second moments). The deviations from $q = 0$ are a signal of diminishing returns epistasis; at this point, let us emphasise again that the approximating curve of the mean relative fitness is (slightly) concave even for $q = 0$ (due to the runtime effect). By any means, the pronounced concavity in the curve approximating the LTEE data (with its estimated $\hat{q} = 4.2$) gives strong evidence for diminishing returns epistasis, in line with the conclusions of previous investigations ([97], [48], [100]). We would like to emphasise, however, that our goal here was not to find the ‘best’ (or even the ‘true’) increment function; rather, the choice (4.11) was made for the sake of comparison with [97], while the GKWY model in fact allows for arbitrary increment functions.

Our third line of investigations is a simulation study both of the Cannings model and the approximating heuristics (the parameter combinations of the study are summarised in Table 4.1). Here it turned out that the heuristics approximates the Cannings model very well (it might be improved even further by taking into account the refined heuristics in [41] and [84]). This suggests that the discrepancy between the (mean of the) Cannings

Law of X	t_{\max}	\hat{q}	$(\hat{\mathfrak{d}}_1)$	$\hat{\mu}$	$\hat{\varphi}$	Algorithm 6	Algorithm 7
$\equiv 1$	7600	4.2	(0.14)	0.035	0.140	Figure 4.4	
$\equiv 1$	7600	4.2	(0.14)	0.079	0.140	Figure 4.5	Figure 4.6
Exp(1)	7600	4.2	(0.20)	0.106	0.069	Figure 4.7	Figure 4.8
shifted Pareto, $\lambda = 2.5$, cf. (4.45)	7600	4.2	(0.12)	0.373	0.020	Figure 4.9	Figure 4.10

Table 4.1.: Summary of parameter estimates and corresponding simulation results. The population size and the dilution factor have been fixed as $N = 5 \cdot 10^6$ and $\gamma = 100$ throughout.

simulations and the approximating power law is mainly due to moment closure. The simulations show that this deviation is moderate for deterministic increments, minute for exponential increments, and hard to quantify for Pareto increments due to the large fluctuations.

Appendix A.

Constrained optimisation

This chapter gives a short recapulation about nonlinear constrained optimisation and a very general solution algorithm for this kind of problems. It is to be considered as a basis to embed the central optimisation problem of Section 3.2 and its numerical evaluation of Section 3.6 in the context of constrained optimisation and to give the reader unfamiliar with this theory a brief access to it.

In Section A.1 the basic theory about this problem class including its necessary and sufficient conditions for identifying local optima for a given problem are summarised. Section A.2 describes the basic ideas of one of the very common general solution algorithms for nonlinear constrained optimisation problems, which is used to estimate the recombination parameters of the Moran model with recombination as discussed in Chapter 3.

A.1. Theory of constrained optimisation

In the unconstrained case of minimising a sufficient smooth function $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ the requirements x^* has to fulfil to be a stationary point are zero slope on the one hand and non-negative curvature in any direction at x^* on the other hand. In the constrained case x^* first of all has to be feasible with respect to some constraints which makes the requirements a little more elaborate. This section provides the necessary nomenclature and theory in a nutshell. More detailed instructions are available f.e. in [37], [43] and [73]. Concerning the duality relation in nonlinear optimisation the reader is referred to [20].

From now on $v \in \mathbb{R}^d$ is considered as a column vector with i -th component v_i . The gradient and Hessian of $f : \mathbb{R}^d \rightarrow \mathbb{R}$ shall be denoted by $\nabla f \in \mathbb{R}^d$ and $\nabla^2 f \in \mathbb{R}^{d \times d}$; consequently the Jacobian of $c : \mathbb{R}^d \rightarrow \mathbb{R}^m$ by $\nabla c \in \mathbb{R}^{m \times d}$ with i -th row containing $(\nabla c_i)^T$.

A (*constrained*) *nonlinear optimisation problem* (NLP) is generally formulated as

$$\min_{x \in \mathbb{R}^{n_x}} f(x) \tag{A.1a}$$

$$\text{s.t. } c_i(x) = 0, \quad i \in E, \tag{A.1b}$$

$$c_i(x) \geq 0, \quad i \in I, \tag{A.1c}$$

where the *objective* $f: \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ and *constraints* $c_i: \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ are sufficiently smooth.²⁹ Let E and I be the finite and disjoint indexsets for *equality* and *inequality constraints*, respectively, and thus the vectors of equalities and inequalities can be denoted as

$$c_E: \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_E}, \quad \text{with} \quad c_E(x) = 0, \quad n_E = |E|, \quad \text{and} \quad (\text{A.2a})$$

$$c_I: \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_I}, \quad \text{with} \quad c_I(x) \geq 0, \quad n_I = |I|. \quad (\text{A.2b})$$

Any point x satisfying (A.2) is called *feasible*,

$$\mathbb{F} = \{x \in \mathbb{R}^{n_x} \mid c_E(x) = 0, \quad c_I(x) \geq 0\} \quad (\text{A.3})$$

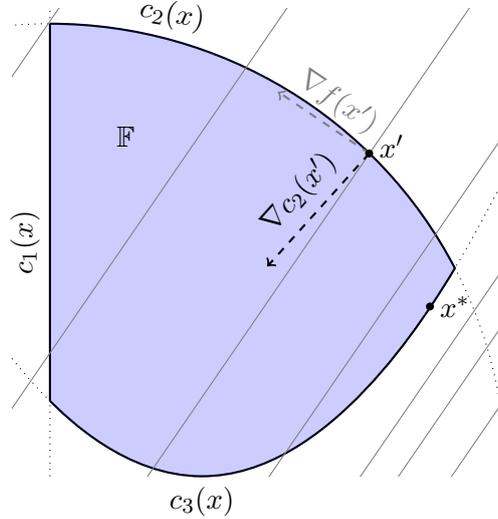
is referred to as the *feasible region* or *feasible set* of (A.1), and a (*local*) *solution* thereof is denoted by x^* .

Another indexset relevant and associated to a NLP, that *depends* on a feasible point, is the *active set*

$$A(x) = E \cup \{i \in I \mid c_i(x) = 0\}. \quad (\text{A.4})$$

Beyond playing a central role in characterising and identifying local solutions, the active set poses as the eponym of one of the chosen subsolvers at the end of Section A.2. Figure A.1 illustrates a NLP and the corresponding, recently introduced (index) sets.

Figure A.1: Showcase for a constrained nonlinear optimisation problem with interior \mathbb{F}° (blue), $\partial\mathbb{F}$ (black), constraints c_1, c_2, c_3 (dotted lines) and level curves of f (grey lines). $I = \{1, 2, 3\}$, $E = \emptyset$, $A(x') = \{2\}$ and $A(x^*) = \{3\}$. $\nabla c_2(x')$ is normal to the contour $c_2 = 0$ with direction on the feasible side, since $x' \in c_2^{-1}(0)$ and $2 \in I$.



Lagrangian duality in mathematical optimisation. Informally, the basic concept of constrained optimisation is to subsume the NLP in standard form (A.1) into one function

²⁹This problem class is sometimes denoted additionally as *continuous* to be more specific and to distinguish it from *discrete optimisation*, where x is drawn from a finite set and from the mixed type class called (*mixed*) *integer programming*, which allows some components of x to take only integer values. The continuous class is considered here because the rest is beyond the scope of application for this thesis.

that augments the objective with a weighted sum of the constraints and seek a stationary point of this auxiliary function.

Definition 4 (Lagrangian function). *Let $\lambda_E = (\lambda_i)_{i \in E} \in \mathbb{R}^{n_E}$ and $\lambda_I = (\lambda_i)_{i \in I} \in \mathbb{R}^{n_I}$ be the vectors of so-called Lagrange multipliers (or dual variables or dual multipliers) for equality and inequality constraints. The function $\Lambda: \mathbb{R}^{n_x} \times \mathbb{R}^{n_E} \times \mathbb{R}^{n_I} \rightarrow \mathbb{R}$ given by*

$$\Lambda(x, \lambda_E, \lambda_I) = f(x) - \sum_{i \in E} \lambda_i c_i(x) - \sum_{i \in I} \lambda_i c_i(x) \quad (\text{A.5})$$

is then called the Lagrangian Function of (A.1) with domain $\mathbb{F} \times \mathbb{R}^{n_E} \times \mathbb{R}^{n_I}$.

The terminology of *dual* variables or multipliers in Definition 4 comes from the duality principle in mathematical optimisation that allows to view the optimisation problem from two perspectives via the Lagrangian function. Namely, defining the *Lagrange dual function* $q(\lambda_E, \lambda_I) = \inf_x \Lambda(x, \lambda_E, \lambda_I)$ it obviously yields a lower bound of $f(x^*)$ for all $\lambda_I \geq 0$, albeit very meaningless when $q(\lambda_E, \lambda_I) = -\infty$. Nevertheless q provides a parametrisation of lower bounds of the original problem that may be asked to be optimised by

$$\max_{\lambda_E, \lambda_I} q(\lambda_E, \lambda_I) \quad \text{s.t.} \quad \lambda_I \geq 0. \quad (\text{A.6})$$

This is called the *dual problem* in *dual variables* to the *primal problem* in (A.1) with *primal variables* x .³⁰ The difference $f(x^*) - q(\lambda_E^*, \lambda_I^*)$ is called the *duality gap* and *strong (weak) duality* holds if the duality gap is (not) equal to zero. For a detailed study about duality in optimisation, and especially its application in the convex case, the interested reader is referred to [20].

Optimality conditions. We give a summary of optimality conditions for a solution of (A.1). These conditions rely essentially on first- and second-order Taylor expansions of f and the constraints about some feasible point $x \in \mathbb{F}$, and answer the question if a feasible descent direction exists in x with respect to this approximation.

Definition 5 (Linearised feasible directions). *Given a feasible point $x \in \mathbb{F}$ and its associated active set $A(x)$ of (A.1). Then $\mathbb{D}(x) = \{d \in \mathbb{R}^{n_x} \mid d^T \nabla c_i(x) = 0 \ \forall i \in E, d^T \nabla c_i(x) \geq 0 \ \forall i \in I \cap A(x)\}$ is called the set of linearised feasible directions.*

Definition 6 (Linear independence constraint qualification (LICQ)). *Given a feasible point $x \in \mathbb{F}$ and its associated active set $A(x)$ of (A.1) the linear independence constraint qualification (LICQ) holds if the set of active constraint gradients $\{\nabla c_i(x), i \in A(x)\}$ is linear independent.*

The feasible set is a geometric object represented by (A.1b) and (A.1c). It can be shown that the tangent cone in $x \in \mathbb{F}$, which is a geometric object as well, is a subset of $\mathbb{D}(x)$, where

³⁰Using the Lagrangian function is not the only way of establishing a dual problem in mathematical optimisation, but the only one considered in this thesis.

the two objects coincide if the LICQ is fulfilled. So the LICQ add additional regularity to the constraints for the local characterisation of \mathbb{F} in x . Trivially but worth to be mentioned is the fact that under the LICQ no active constraint gradient is zero.³¹

Theorem 3 (Karush-Kuhn-Tucker conditions). *Let $x^* \in \mathbb{R}^{n_x}$ be a local solution of (A.1) and let f and c_i be continuously differentiable for all $i \in E \cup I$. In addition, assume that the LICQ holds at x^* . Then, there exist vectors $\lambda_E^* \in \mathbb{R}^{n_E}$ and $\lambda_I^* \in \mathbb{R}^{n_I}$ such that the following conditions*

$$c_E(x^*) = 0, \quad (\text{A.7a})$$

$$c_I(x^*) \geq 0, \quad (\text{A.7b})$$

$$\nabla f(x^*) - \sum_{i \in E} \lambda_i^* \nabla c_i(x^*) - \sum_{i \in I \cap A(x^*)} \lambda_i^* \nabla c_i(x^*) = 0, \quad (\text{A.7c})$$

$$\lambda_I^* \geq 0, \quad (\text{A.7d})$$

$$\lambda_i^* c_i(x^*) = 0, \quad i \in E \cup I, \quad (\text{A.7e})$$

are satisfied. $(x^*, \lambda_E^*, \lambda_I^*)$ is called a stationary point (or KKT point) of problem (A.1).

As a short survey of possibilities to interpret the conditions: (A.7a) together with (A.7b) ensure the *primal feasibility*, i.e. x^* is a feasible point of the primal problem. Traditionally, (A.7c) is also called *stationarity condition*, since it claims $\nabla_x \Lambda(x^*, \lambda_E^*, \lambda_I^*) = 0$. Together with (A.7d) these two conditions ask the *dual feasibility*, i.e. $(\lambda_E^*, \lambda_I^*)$ to be a feasible point of the dual problem. Moreover, due to (A.7e) by itself, $f(x^*) = \Lambda(x^*, \lambda_E^*, \lambda_I^*)$ holds.

So far Theorem 3 yields necessary conditions for a KKT point that take into account first derivative informations and describe the relation between them. Yet they are not sufficient to decide whether a direction $w \in \mathbb{D}(x^*)$ will increase or decrease the objective function. This can be improved by taking informations of second-order terms of the Taylor series expansions of f and c_i into account together with a relevant subcone of \mathbb{D} , that is given by

$$\nabla c_i(x^*)^T w = 0, \quad i \in E, \quad (\text{A.8a})$$

$$\nabla c_i(x^*)^T w = 0, \quad i \in I \cap A(x^*) \text{ with } \lambda_i^* > 0, \quad (\text{A.8b})$$

$$\nabla c_i(x^*)^T w \geq 0, \quad i \in I \cap A(x^*) \text{ with } \lambda_i^* = 0, \quad (\text{A.8c})$$

and often referred to as the *critical cone* at a KKT point.

Theorem 4 (Second-order necessary conditions). *Let the assumptions of Theorem 3 hold and $(x^*, \lambda_E^*, \lambda_I^*)$ be the resulting KKT point, then*

$$w^T \nabla_{xx}^2 \Lambda(x^*, \lambda_E^*, \lambda_I^*) w \geq 0 \quad (\text{A.9})$$

holds for all w in the critical cone.

³¹The LICQ is not the only constraint qualification proposed in the literature. They all have in common to maintain the similarity of the tangent cone and the set of linearised feasible directions.

The corresponding sufficient second-order conditions ensuring that a x^* is local solution of the primal problem, then, look very similar to Theorem 4 and can even neglect the LICQ. Fulfilling (A.9) strictly identifies even a *strict* local solution.

Theorem 5 (Second-order sufficient conditions). *Let $(x^*, \lambda_E^*, \lambda_I^*)$ be a KKT point by the means of Theorem 3 with*

$$w^T \nabla_{xx}^2 \Lambda(x^*, \lambda_E^*, \lambda_I^*) w > 0 \quad (\text{A.10})$$

for all $w \neq 0$ in the corresponding critical cone, then x^ is a strict local solution of (A.1).*

A.2. Numerical algorithm for constrained optimisation

Global solutions are usually very hard to identify, except from some special cases where local solution are global, like f.e. convex problems. Therefore, most solution algorithm are designed to find a local minimum of a NLP. This section introduces some of the concepts to construct algorithms that are able find a local solution of (A.1) iteratively, i.e. by generating a sequence $\{x^{(k)}\}$ of estimates of x^* converging to it. It follows the way of [15], supplemented by [73], to establish a foundation to understand the basic concepts, obstacles, and functionality of a *sequential quadratic programming* algorithm (SQP) that is stated at the end of this section and used in Chapter 3. For a more detailed insight on the interplay of the building blocks see also [82]. A more elaborated explanation and description of the broad class of SQP algorithms is surveyed in [44] and for even alternative algorithmic approaches like *Interior Point Methods* the reader is again referred to [73].

Newton method for equality constrained optimisation. At first, consider the case when only equality constraints are involved, i.e. $I = \emptyset$. Then, the NLP and its necessary and sufficient KKT conditions at a local solution read

$$\min_{x \in \mathbb{R}^{n_x}} f(x) \quad \text{s.t.} \quad c_E(x) = 0, \quad (\text{A.11a})$$

$$\nabla_x \Lambda(x^*, \lambda_E^*) = \nabla f(x^*) - (\nabla c_E(x^*))^T \lambda_E^* = 0, \quad c_E(x^*) = 0, \quad (\text{A.11b})$$

$$p^T \nabla_{xx}^2 \Lambda(x^*, \lambda_E^*) p > 0 \quad \forall p \neq 0, \quad (\nabla c_E(x^*))^T p = 0. \quad (\text{A.11c})$$

The principle strategy to calculate a solution of (A.11a) consists of solving the nonlinear system (A.11b) under the assumption (A.11c) and LICQ, at which the latter means $\nabla c_E(x^*)$ having full row rank. Therefore a strategy based on Newton's method is pursued that takes generations of Newton steps from the linear system

$$\begin{bmatrix} \nabla_{xx}^2 \Lambda^{(k)} & (C_E^{(k)})^T \\ C_E^{(k)} & 0 \end{bmatrix} \begin{pmatrix} p_x \\ -p_{\lambda_E} \end{pmatrix} = - \begin{pmatrix} \nabla f^{(k)} - (C_E^{(k)})^T \lambda_E^{(k)} \\ c_E^{(k)} \end{pmatrix} \quad (\text{A.12})$$

with $\nabla_{xx}^2 \Lambda^{(k)} = \nabla_{xx}^2 \Lambda(x^{(k)}, \lambda_E^{(k)})$, $C_E^{(k)} = \nabla c_E(x^{(k)})$, $\nabla f^{(k)} = \nabla f(x^{(k)})$ and $c_E^{(k)} = c_E(x^{(k)})$. By using $\lambda_E^{(k+1)} = \lambda_E^{(k)} + p_{\lambda_E} = \lambda_{QP}$ this linear system is equivalent to

$$\begin{bmatrix} \nabla_{xx}^2 \Lambda^{(k)} & (C_E^{(k)})^T \\ C_E^{(k)} & 0 \end{bmatrix} \begin{pmatrix} p_x \\ -\lambda_{QP} \end{pmatrix} = - \begin{pmatrix} \nabla f^{(k)} \\ c_E^{(k)} \end{pmatrix}, \quad (\text{A.13})$$

which, in turn, constitutes the necessary KKT conditions of a *quadratic programming problem (QP)*

$$\min_{p \in \mathbb{R}^{n_x}} \frac{1}{2} p^T \nabla_{xx}^2 \Lambda^{(k)} p + p^T \nabla f^{(k)} \quad \text{s.t.} \quad C_E^{(k)} p + c_E^{(k)} = 0. \quad (\text{A.14})$$

The matrix in (A.13) is called *Karush-Kuhn-Tucker matrix* (KKT matrix), consequently.

Algorithm 8: Basic Newton method for equality constrained NLP

Input: User provided starting point $(x^{(0)}, \lambda_E^{(0)})$ for problem (A.11a).

- 1 Set iteration counter $k = 0$.
 - 2 Evaluate $\nabla_{xx}^2 \Lambda^{(0)}$, $C_E^{(0)}$, $\nabla f^{(0)}$, and $c_E^{(0)}$ at $(x^{(0)}, \lambda_E^{(0)})$.
 - 3 **while** $\|p_x^{(k)}\|, \|p_{\lambda_E}^{(k)}\| < \epsilon_1$, $\|\nabla_x \Lambda(x^{(k)}, \lambda_E^{(k)})\| < \epsilon_2$ **does not hold do**
 - 4 **if** KKT matrix is singular **then**
 - 5 | Stop.
 - 6 Solve (A.13) to obtain p_x and λ_{QP} .
 - 7 Set $x^{(k+1)} = x^{(k)} + p_x$ and $\lambda_E^{(k+1)} = \lambda_{QP}$.
 - 8 Evaluate $\nabla_{xx}^2 \Lambda^{(k+1)}$, $C_E^{(k+1)}$, $\nabla f^{(k+1)}$, and $c_E^{(k+1)}$ at $(x^{(k+1)}, \lambda_E^{(k+1)})$.
 - 9 Increase iteration counter $k \leftarrow k + 1$.
 - 10 **return** optimal solution $(x^{(k)}, \lambda_E^{(k)})$.
-

Theorem 6. *Algorithm 8 converges quadratically in a neighbourhood of (x^*, λ_E^*) satisfying (A.11c) and the LICQ, if $\nabla^2 f(x)$ and $\nabla^2 c_E(x)$ are Lipschitz continuous therein.*

The proof of Theorem 6 follows the proof of Newton's method for unconstrained optimisation but relies additionally on the nonsingularity of the KKT matrix. Thus this property has to be guaranteed in line 4 for the algorithm to succeed. The following theorem for general QPs with Hessian matrix G and constraint matrix C provides direct conditions under which a unique solution (x^*, λ_E^*) of its KKT system exists with x^* being the *global* optimum.

Theorem 7. *Let $C \in \mathbb{R}^{m \times n}$ have full row rank and let Z denote the $n \times (n - m)$ null space matrix of C , i.e. $CZ = 0$. Assume that the reduced-Hessian matrix $Z^T G Z$ is positive definit. Then the following statements hold.*

1. *The corresponding KKT matrix is nonsingular.*
2. *There exists a unique vector that satisfies the KKT system and is global solution of the corresponding QP.*

The nonsingular KKT system may be solved by any direct (factoring, Schur complement, null-space) or iterative (CG and its derivatives) method from numerical linear algebra. For a discussion of theirs (dis-)advantages as well as a proof of Theorem 7 see [73].

Globalisation through step acceptance by the use of a filter concept. Algorithms solving NLPs generate steps from one iterate $x^{(k)}$ to the next $x^{(k+1)}$ usually either through a *line search* or *trust region* approach. In either *globalisation strategy* first a *direction* is computed with the help of a model problem and the subsequent determination of a trial step $x^{(+)} = x^{(k)} + \alpha p_x$ is provided with another degree of freedom by the *step size* $\alpha \in (0, 1]$. Thus trial points may be infeasible but nevertheless shall make significant algorithmic progress and, by that, be a good intermediate step on the way to the solution x^* , especially in the case when the current iterate is not sufficiently close to the solution. So being in a current primal point $x^{(k)}$ a globalisation strategy has to consider the two (often competing) goals of

- reducing of the objective function and
- satisfying the constraints

of the original NLP. The two central approaches are *merit functions* on the one hand and *filter methods* on the other hand. Both will accept a step size α only if the balance between these goals is preserved. Merit functions $\phi(x, \mu)$ take the objective of (A.11a) or its Lagrangian function, add a weighted (by μ) penalisation of the constraints violation, and determine the *step length* in direction p such that a *sufficient decrease* of the merit function is reached.

Since in this thesis a SQP filter line search approach is used, now a more elaborate description of the latter is given. In contrast to merit functions a filter keeps the two above-mentioned goals separate. If we define a measure of constraints violation like

$$\tilde{c}(x) = \|c_E(x)\|, \quad (\text{A.15})$$

the filter concept is described by Definition 7 and illustrated in Figure A.2, cf. [73].

Definition 7 (Acceptance and domination of filter pairs).

1. A pair $(f^{(k)}, \tilde{c}^{(k)})$ is said to dominate another pair $(f^{(j)}, \tilde{c}^{(j)})$ if both $f^{(k)} \leq f^{(j)}$ and $\tilde{c}^{(k)} \leq \tilde{c}^{(j)}$.
2. A filter is a list of pairs $(f^{(j)}, \tilde{c}^{(j)})$ such that no pair dominates any other.
3. An iterate $x^{(k)}$ is said to be acceptable by the filter if $(f^{(k)}, \tilde{c}^{(k)})$ is not dominated by any pair in the filter.

A backtracking filter line search algorithm accepts a trial step

$$x^{(+)} = x^{(k)} + \alpha^{(k,l)} p_x \quad (\text{A.16})$$

as a new iterate $x^{(k+1)} := x^{(+)}$ if the pair $(f^{(+)}, \tilde{c}^{(+)}) := (f(x^{(+)}, \tilde{c}(x^{(+)}))$ is not dominated by a pair $(f^{(j)}, \tilde{c}^{(j)})$ that is already an element of the filter because of a previous iterate $x^{(j)}$. Consequently, the pair $(f^{(+)}, \tilde{c}^{(+)})$ is added to the filter and all dominated pairs are removed. Otherwise a subsequent backtracking line search is executed to reduce the stepsize $\alpha^{(k,l)}$, $l = 0, 1, \dots$, with $\lim_l \alpha^{(k,l)} = 0$ until an acceptable trial point is found. This approach is enhanced by several techniques to ensure global convergence and good performance. They are traced in the following leading to Algorithm 9. See [91] and [92] for an exhaustive consideration and especially [82] for the description of an initialisation strategy (cf. line 1 in Algorithm 9), which is omitted here.

Sufficient reduction. First, the criterion of acceptance in Definition 7 is tightened, whereby it is avoided to accept points $x^{(+)}$ whose associated pair $(f^{(+)}, \tilde{c}^{(+)})$ is too close to the dominated region of the current filter with emphasis on the constraints violation. So based on the current iterate $x^{(k)}$ and its pair $(f^{(k)}, \tilde{c}^{(k)})$ a derived trial point $x^{(+)}$ is accepted if its pair is outside the dominated region *and* a sufficient improvement based on a small fraction of the current infeasibility $\tilde{c}^{(k)}$ is achieved, i.e.

$$f^{(+)} \leq f^{(k)} - \gamma_f \tilde{c}^{(k)} \quad \text{or} \quad \tilde{c}^{(+)} \leq (1 - \gamma_{\tilde{c}}) \tilde{c}^{(k)} \quad (\text{A.17})$$

for $\gamma_f, \gamma_{\tilde{c}} \in (0, 1)$ and typically small. In particular for almost feasible points $x^{(k)}$ sufficient reduction of the objective is required. This is enforced by a *f-type switching condition*

$$p_x^T (\nabla_x f^{(k)}) < 0 \quad \text{and} \quad \alpha^{(k,l)} \left[-p_x^T (\nabla_x f^{(k)}) \right]^{\kappa_f} > \kappa_{\delta} \left[\tilde{c}^{(k)} \right]^{\kappa_{\tilde{c}}} \quad (\text{A.18})$$

with constants $\kappa_{\delta} > 0, \kappa_{\tilde{c}} > 1, \kappa_f > 2\kappa_{\tilde{c}}$. Eq. (A.18) holds for a descent direction p_x with respect to f that provides sufficient progress for the objective function compared to the constraints violation for a step size $\alpha^{(k,l)}$ fulfilling the Armijo condition

$$f^{(+)} \leq f^{(k)} + \kappa_{\alpha} \alpha^{(k,l)} p_x^T (\nabla_x f^{(k)}) \quad (\text{A.19})$$

for a $\kappa_{\alpha} \in (0, 0.5)$.

Feasibility restoration phase. Unfortunately, it is not always guaranteed, to find a step size desired by the above criteria. Then a feature called *feasibility restoration phase* is applied whose purpose is to find a new iterate $x^{(k+1)} = x^{(+, \text{frp})}$ that is acceptable for the current filter and satisfies (A.18). Any iterative algorithm that tries to find a less infeasible point can be used, f.e. by minimising \tilde{c} without considering the objective at all. Rose [82] suggests an approach that determines $(x^{(+, \text{frp})}, \lambda^{(+, \text{frp})})$ with the additional property of

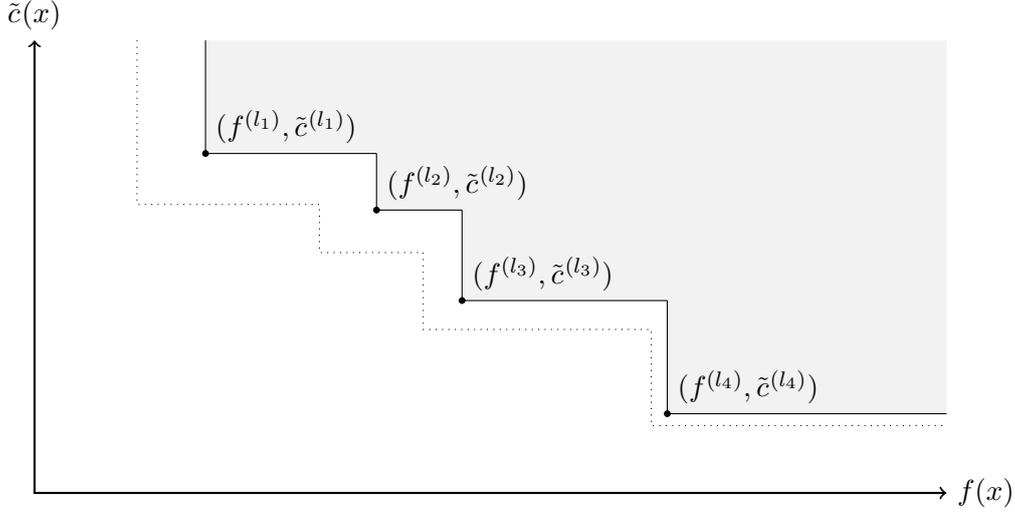


Figure A.2.: An illustration of the filter concept. The filter contains the pairs $(f^{(l_i)}, \tilde{c}^{(l_i)})$, $i = 1, 2, 3, 4$ and the dominated area with respect to Definition 7 is highlighted in grey, i.e. right and above the (including) border determined by the filter pairs. When using the tighter criterion (A.18) the arising border is indicated by the dotted line.

being not too far away from $(x^{(k)}, \lambda_E^{(k)})$. To detect this situation the linear approximations

$$f^{(+)} \approx f^{(k)} + \alpha p_x^T (\nabla_x f^{(k)}) \quad \text{and} \quad \tilde{c}^{(+)} \approx \tilde{c}^{(k)} - \alpha \tilde{c}^{(k)}$$

applied to (A.17) together with (A.18) yield a lower bound

$$\alpha_{\min}^{(k)} = \gamma_\alpha \begin{cases} \min \left\{ \kappa_{\tilde{c}}, \frac{\gamma_f \tilde{c}^{(k)}}{-p_x^T (\nabla_x f^{(k)})}, \frac{\kappa_\delta [\tilde{c}^{(k)}]^{\kappa_{\tilde{c}}}}{[-p_x^T (\nabla_x f^{(k)})]^{\kappa_f}} \right\}, & p_x^T (\nabla_x f^{(k)}) < 0 \\ \kappa_{\tilde{c}}, & \text{otherwise} \end{cases} \quad (\text{A.20})$$

for $\alpha^{(k,l)}$ with $\gamma_\alpha \in (0, 1]$.

Second order correction. The filter approach can suffer from the *Maratos effect*, i.e. a full Newton step $\alpha^{(k)} = 1$ increases *both* f and \tilde{c} , which is even unfavourable close to a local solution of (A.11a) and leading to much poorer convergence performance than expectable by a Newton method; cf. Figure A.3 for an illustration. The *second order correction* (soc) approach tries to overcome this behaviour by determining another Newton search direction p_x^{soc} for the constraints evaluated at the point $x^{(k)} + p_x$ and check the conditions (A.17) or (A.18), respectively, for the soc-trial point $x^{(+,\text{soc})} = x + p_x + p_x^{\text{soc}}$ whenever the full step p_x is not accepted. Its algorithmic filter acceptance criteria are similar to those of $x^{(+)}$ in (A.17), (A.18) and (A.19). The details are expounded in [91] but omitted here and in Algorithm 9.

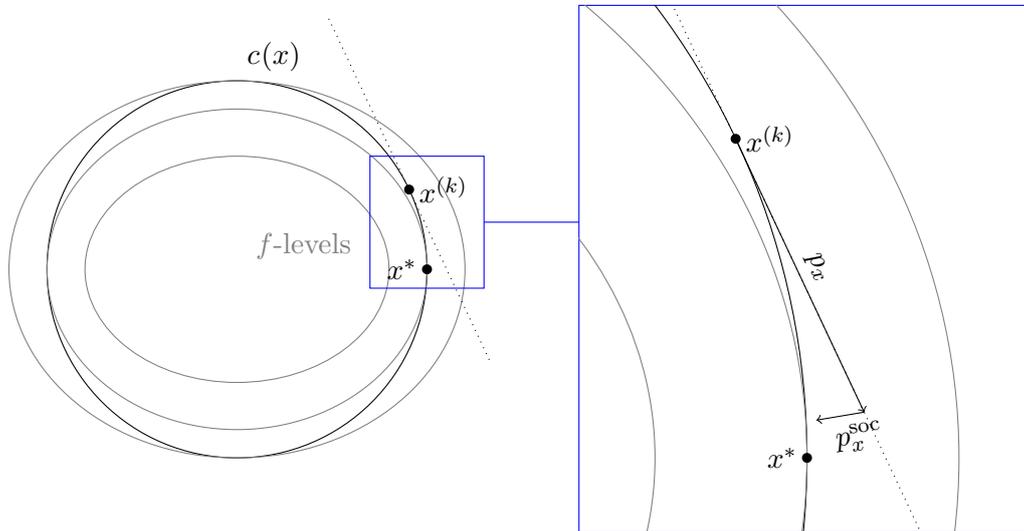


Figure A.3.: Idea of second order correction. In this situation, the current QP approximating the NLP at the (already) feasible point $x^{(k)}$ determines the direction p_x . A step of size $\alpha^{(k,0)} = 1$ along p_x would lead to a point, which violates the constraints again and even increases the objective. However, a subsequent step along p_x^{soc} remedies the problem.

Description	Symbol	Range	Default
Filter margin f	γ_f	$(0, 1)$	10^{-5}
Filter margin \tilde{c}	$\gamma_{\tilde{c}}$	$(0, 1)$	10^{-5}
Switching condition factor	κ_{δ}	> 0	1
Switching condition factor	$\kappa_{\tilde{c}}$	> 1	1.1
Switching condition factor	κ_f	$> \kappa_{\tilde{c}}$	2.3
Armijo factor	κ_{α}	$(0, 0.5)$	10^{-4}
Safety guard for α_{\min}	γ_{α}	$(0, 1]$	1
Backtracking factor	κ_b	$(0, 1)$	0.5

Table A.1.: Algorithmic constants of a filter line-search SQP algorithm; defaults from [82].

Algorithm 9: Filter line-search SQP algorithm

Input : Initial point $\bar{x}^{(0)}$, vector of algorithmic constants κ (see Table A.1).

- 1 Use an initialisation strategy for the filter and to obtain $(x^{(0)}, \lambda_E^{(0)})$ from $\bar{x}^{(0)}$.
- 2 Set outer iteration counter $k = 0$.
- 3 Evaluate NLP data $\nabla_{xx}^2 A^{(0)}$, $C_E^{(0)}$, and $\nabla f^{(0)}$, $c_E^{(0)}$ at $(x^{(0)}, \lambda_E^{(0)})$.
- 4 **while** *NLP termination criterion* (A.11b) **does not hold do**
 - // Determination of step direction.
 - 5 Solve QP^(k) (A.13) to obtain p_x and λ_{QP} .
 - 6 **if** *QP^(k) is ill-conditioned or singular* **then**
 - 7 | Go to feasibility restoration phase, line 29.
 - // Backtracking line search.
 - 8 Set $\alpha^{(k,0)} = 1$ and $l \leftarrow 0$.
 - 9 **if** $\alpha^{(k,l)} < \alpha_{\min}^{(k)}$ *w.r.t. (A.20) holds* **then**
 - 10 | Go to feasibility restoration phase, line 29.
 - 11 Set $x^{(+)} = x^{(k)} + \alpha^{(k,l)} p_x$.
 - 12 **if** $(f^{(+)}, \tilde{c}^{(+)})$ *is dominated by the current filter* **then**
 - 13 | Reject $\alpha^{(k,l)}$ and go to line 21.
 - 14 **if** *f-type switching condition* (A.18) **and** *Armijo condition* (A.19) **holds then**
 - 15 | Accept $\alpha^{(k)} \leftarrow \alpha^{(k,l)}$, the filter remains unaffected and go to line 24.
 - 16 **else if** (A.18) *fails but sufficient reduction w.r.t. (A.17) is achieved* **then**
 - 17 | Accept $\alpha^{(k)} \leftarrow \alpha^{(k,l)}$ and augment the filter with $(f^{(k)} - \gamma_f \tilde{c}^{(k)}, (1 - \gamma_{\tilde{c}}) \tilde{c}^{(k)})$.
 - 18 | Go to line 24.
 - 19 **else**
 - 20 | Go to line 21.
 - // Second order correction.
 - 21 **if** $l = 0$ **and** *a subsequent second order correction is successful* **then**
 - 22 | Accept $p_x \leftarrow p_x + p_x^{\text{soc}}$, $\alpha^{(k)} = \alpha^{(k,0)}$ and go to line 24.
 - // Update stepsize.
 - 23 Update $\alpha^{(k,l)} \leftarrow \kappa_b \alpha^{(k,l)}$, $l \leftarrow l + 1$ and go to line 9.
 - // Update KKT point iterate and NLP data.
 - 24 Make step $x^{(k+1)} \leftarrow x^{(k)} + \alpha^{(k)} p_x$ and $\lambda_E^{(k+1)} \leftarrow (1 - \alpha^{(k)}) \lambda_E^{(k)} + \alpha^{(k)} \lambda_{QP}$.
 - 25 Evaluate $\nabla_{xx}^2 A^{(k+1)}$, $C_E^{(k+1)}$, and $\nabla f^{(k+1)}$, $c_E^{(k+1)}$ at $(x^{(k+1)}, \lambda_E^{(k+1)})$.
 - 26 Evaluate the KKT error at $(x^{(k+1)}, \lambda_E^{(k+1)})$ by the means of (A.11b).
 - 27 Increase iteration counter $k \leftarrow k + 1$.
- 28 **return** optimal solution $(x^{(k)}, \lambda_E^{(k)})$.
- // Feasibility restoration phase
- 29 Obtain a new iterate $x^{(+, \text{frp})}$ that is acceptable by the current filter.
- 30 Augment the filter with $(f^{(k)} - \gamma_f \tilde{c}^{(k)}, (1 - \gamma_{\tilde{c}}) \tilde{c}^{(k)})$.
- 31 Set $x^{(k+1)} \leftarrow x^{(+, \text{frp})}$, $\lambda_E^{(k+1)} \leftarrow \lambda^{(+, \text{frp})}$ and go to line 26.

Sequential quadratic programming for general NLP. Closing this overview, Algorithm 9 is extended to solve nonlinear optimisation problems that may also include inequality constraints and thus are of the form (A.1). Considering the QP (A.14) as a (local) quadratic model with first order approximation of the (equality) constraints, in the general case the inequality constraints are linearised the same way leading to the more general QP

$$\min_{p \in \mathbb{R}^{n_x}} \quad \frac{1}{2} p^T \nabla_{xx}^2 \Lambda^{(k)} p + p^T \nabla f^{(k)} \quad (\text{A.21a})$$

$$\text{s.t.} \quad C_E^{(k)} p + c_E^{(k)} = 0, \quad (\text{A.21b})$$

$$C_I^{(k)} p + c_I^{(k)} \geq 0, \quad (\text{A.21c})$$

corresponding to (A.1) and (A.5) in $x^{(k)}$ with $C_I^{(k)} = \nabla c_I(x^{(k)})$ and $c_I^{(k)} = c_I(x^{(k)})$. $\nabla_{xx}^2 \Lambda^{(k)}$ is even allowed to be replaced by a symmetric matrix approximation. Then, in Algorithm 9, the following slight adjustments have to be implemented.

- The calculation of a search direction in line 5 is replaced by solving (A.21) requiring all associated NLP data evaluations beforehand at the KKT point iterate $(x^{(k)}, \lambda_E^{(k)}, \lambda_I^{(k)})$. Since in a neighborhood of the NLP solution x^* the active sets $A(x^{(k)})$ of consecutive iterates are almost the same an *elastic active-set method* with the possibility to be *warm started* appear to be highly applicable as it is suggested in [82].
- The feasibility restoration phase has to capture the inequality constraints as well.
- The constraints violation $\tilde{c}(x) = \|(c_E(x)^T, (\min\{c_i^{(k)}, 0\})_{i \in I}^T)\|_\infty$ is augmented.
- The termination criterion has to be replaced by (A.7a) (A.7b) (A.7c), verified by

$$\tilde{c}^{(k)} < \epsilon_{\text{prim}} \quad \text{and} \quad \|\nabla_x \Lambda(x^{(k)}, \lambda_E^{(k)}, \lambda_I^{(k)})\|_\infty < \epsilon_{\text{dual}} \quad (\text{A.22})$$

for some so-called *primal* and *dual feasibility tolerances* ϵ_{prim} and ϵ_{dual} . The conditions (A.7d) and (A.7e) are provided by the active set method by returning $p_x = 0$ and suitable $\lambda^* = \lambda_{\text{QP}}$ at a stationary point x^* of (A.1).

The corresponding theorems ensuring local and global convergence properties of line search filter methods for nonlinear programming outlined in this chapter are stated in the likewise entitled publications [91] and [92].

Appendix B.

Multiple shooting approach for parameter estimation problems

Section B.1 and Section B.2 follow the general description of parameter estimation problems and multiple shooting by H. G. Bock, who proposed a boundary value problem approach for parameter estimation problems based on multiple shooting (see [18] and [19]).

B.1. Parameter estimation problem

For the purpose of this thesis a parameter estimation problem is characterised by a system of ODEs for a state variable $u(t)$ that additionally depends on a parameter vector p

$$\dot{u} = f(t, u, p). \quad (\text{B.1})$$

At timepoints $t_0 \leq t_1 < \dots < t_o \leq t_f$ within a finite interval $I = [t_0, t_f]$ measurements or observations for functions of the states u shall be denoted by

$$\eta_{ij} = g_{ij}(u(t_j), p) + \epsilon_{ij}, \quad (\text{B.2})$$

where ϵ_{ij} capture commonly unavoidable measurement/observation errors. Given a (to be) specified optimality criterion, the *inverse problem* consists of the task to determine the unknown parameter p , such that the model, that is related to this vector, captures the observed data η best. One of the most common quality criterion is the weighted l_2 -norm: assuming the errors to be component-wise and independent of time $N(0, \sigma_{ij})$ distributed, the objective function of the inverse problem is given by

$$l_2(u, p) = \sum_{i=1}^o \sum_j \left(\frac{g_{ij}(u(t_i), p) - \eta_{ij}}{\sigma_{ij}} \right)^2 = \sum_{i=1}^o \|\epsilon_{ij}\|_{\Sigma_i^{-1}}^2 \quad (\text{B.3})$$

with $\Sigma_i^{-1} = \text{diag}((\sigma_{ij}^{-2})_j)$.³² For the purpose of simplifying the readability consider the summands in (B.3) as components of a residual vector $\tilde{r} \in \mathbb{R}^{n_r}$, where n_r is the number of all measurements, the parameter estimation problem may be formalised in the following

³²See also [73, Chap. 10] for a short discussion of l_2 and other norms in this context.

canonical manner

$$\min_{u,p} l_2(u, p) = \min_{u,p} \|\tilde{r}(u(t_1), \dots, u(t_o), p)\|_2 \quad (\text{B.4a})$$

$$\text{s.t.} \quad \dot{u} - f(t, u, p) = 0, \quad (\text{B.4b})$$

$$\tilde{c}_E(u(t_1), \dots, u(t_o), p) = 0, \quad (\text{B.4c})$$

$$\tilde{c}_I(u(t_1), \dots, u(t_o), p) \geq 0. \quad (\text{B.4d})$$

Problems of the form (B.4) can be interpreted as a (*constraint*) *multipoint boundary value problems* (MPBVP) and appear in a large domain of mathematical modelling (see e.g. the reference list given in [19, Sec. 7]).

B.2. Multiple shooting approach

Problem (B.4) is an infinite dimensional nonlinear optimisation problem. One possibility to reformulate it as a finite dimensional problem is the so-called direct multiple shooting approach. This section recapitulates its idea of how to discretise and parametrise the states $u(t)$ and thus convert (B.4) into a finite-dimensional NLP of the form (A.1).

Consider a covering grid $[\tau_1, \tau_{n_\tau}] = [t_0, t_f]$ such that

$$\tau_1 < \dots < \tau_{n_\tau}, \quad \Delta_j = \tau_{j+1} - \tau_j, \quad j = 1, \dots, n_\tau - 1, \quad (\text{B.5})$$

and use $j(i) = \arg \min_{j: \tau_j \leq t_i} t_i - \tau_j$ in the following for the ease of notation. Then, introduce a vector $v = (v_1^T, \dots, v_{n_\tau}^T)^T$ of n_τ additional local vector-valued variables for every subinterval $I_j = [\tau_j, \tau_{j+1}]$. Now, the extended vector of variables $x = (v^T, p^T)^T \in \mathbb{R}^{n_\tau n_d + n_p}$ makes it possible to compute the solutions of *independent, local* initial value problems

$$\begin{aligned} \dot{u} &= f(t, u, p), \quad t \in I_j, \\ u(\tau_j) &= v_j \end{aligned} \quad (\text{B.6})$$

for $1 \leq j < n_\tau$ and ‘connect’ them to a (discontinuous) parametrisation of $u(t)$, $t \in I$. Then, the local solution evaluated at the observation timepoints $t_i \in [\tau_{j(i)}, \tau_{j(i)+1})$

$$u(t_i; v_{j(i)}, p), \quad (\text{B.7})$$

are formally inserted in (B.4a), (B.4c), and (B.4d), which leads to

$$r(x) := r(u(t_1; v_{j(1)}, p), \dots, u(t_o; v_{j(o)}, p)), \quad (\text{B.8a})$$

$$c_E(x) := \begin{pmatrix} (u(\tau_{j+1}; v_j, p) - v_{j+1})_{j=1, \dots, n_\tau-1} \\ \tilde{c}_E(u(t_1; v_{j(1)}, p), \dots, u(t_o; v_{j(o)}, p)) \end{pmatrix}, \quad (\text{B.8b})$$

$$c_I(x) := \tilde{c}_I(u(t_1; v_{j(1)}, p), \dots, u(t_o; v_{j(o)}, p)). \quad (\text{B.8c})$$

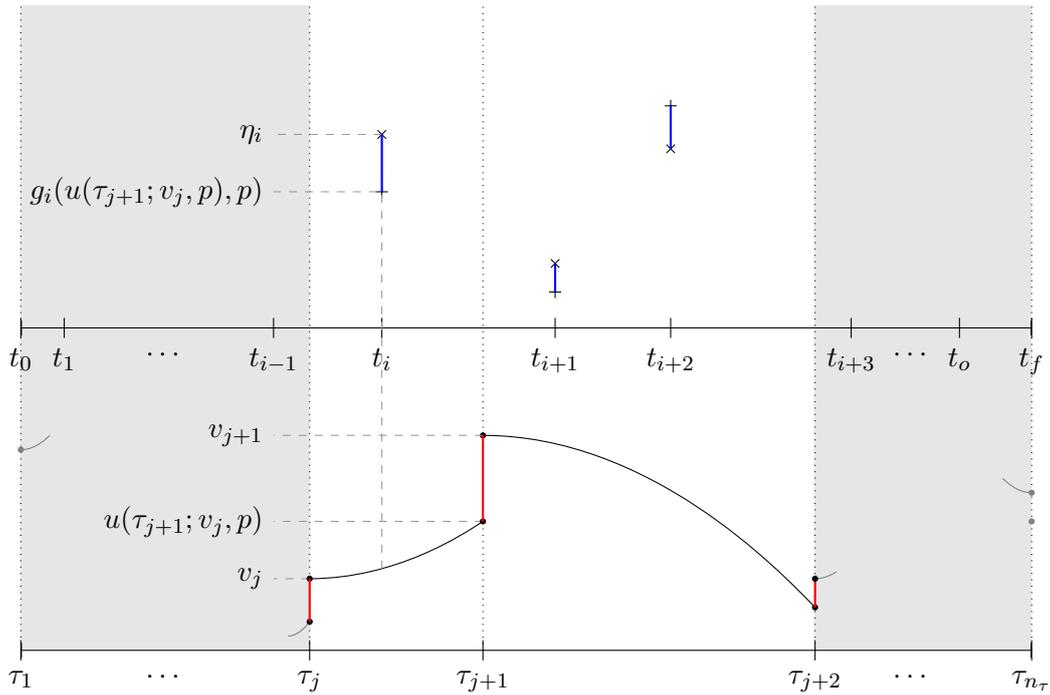


Figure B.1.: Schematic illustration of the multiple shooting approach. The red distance indicates the continuity gap (B.8b), that has to be closed (up to some tolerance); the blue line indicates the residual component of the objective at corresponding measurement times $t_i \in \{t_1, \dots, t_o\}$ that has to be minimized.

Thereby the ODE constraint (B.4b) is exchanged for the additional equality constraints in (B.8b) in every interval end to maintain continuity of the trajectory; and thereby equivalence to (B.4). Figure B.1 illustrates the presented approach.

At the end, a constrained finite-dimensional parameter estimation problem is derived

$$\min_x \|r(x)\|_2 \quad \text{s.t.} \quad c_E(x) = 0, \quad c_I(x) \geq 0, \quad (\text{B.9})$$

that is of the form (A.1). Due to the special objective function it may be denoted as a *constrained discretised parameter least squares boundary value problem*. This approach may be extended to a more complex class of so called optimal control problems, which has a wide field of application nowadays in controlling machines, robots or networks. In this case the objective as well as the constraints in (B.4) may additionally depend on independent control variables from an infinite-dimensional function space. In the multiple-shooting approach they are replaced on a covering grid of $[t_0, t_f]$, which itself should be a superset of $\{\tau_1, \dots, \tau_{n_\tau}\}$, by basis functions (f.e. piecewise polynomial). Then, the finite set of basis function parameters become NLP variables as well.

Finally, it is worth mentioning, that an *infeasible-step* solution algorithm (like SQP in Appendix A) is a reasonable choice for computing a solution of (B.8), since especially the continuity constraint is only fulfilled close to a local solution (up to some tolerances)

and there is a conflict of aims to handle between reducing the objective and reducing the infeasibility, see Figure B.1. On the one hand the residual components $\eta_i - g_i(u(\tau_{j+1}; v_j, p), p)$ (blue) have to be minimised and, on the other hand, the gaps between the endpoints of the local trajectories $u(\tau_j + 1; v_j, p)$ and the initial values v_{j+1} of the next interval have to be closed.

Appendix C.

Supplemental figures for Numerical results in Section 3.6

C.1. Results of experiments regarding different test cases

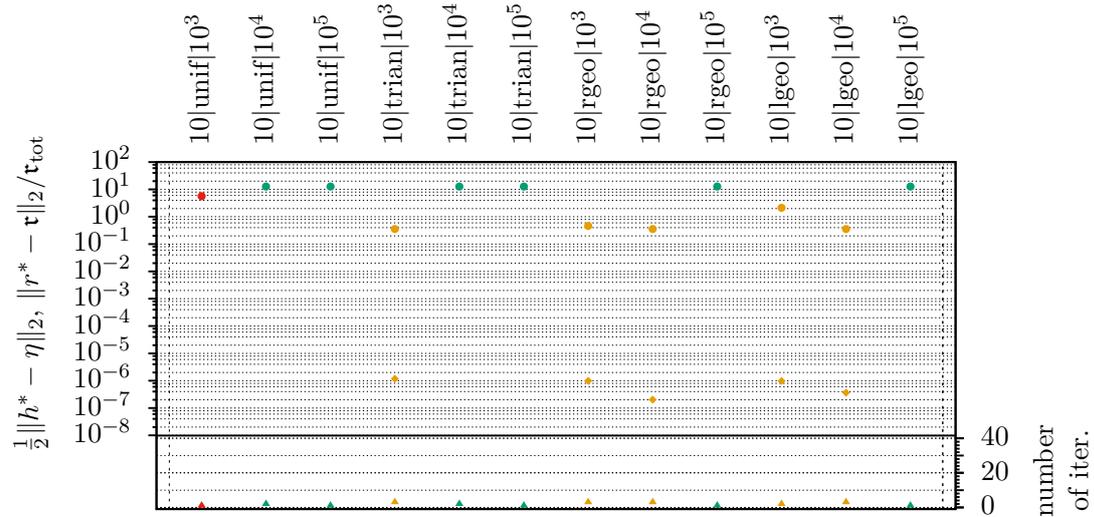


Figure C.1.: Zoom in on the column of Figure C.2 with bottom label $n = 2$ and top label $\tau_{\text{tot}}^{-1} = 10$ showing the results of the experiments regarding `[8000:uniform:6;1.0;full]` applied to the test subset with those fixed values of n and τ_{tot}^{-1} . Top labels: $\tau_{\text{tot}}^{-1}|\text{shape of } \tau|N$. The order of the simulation scenarios used here is the same in all similar figures.

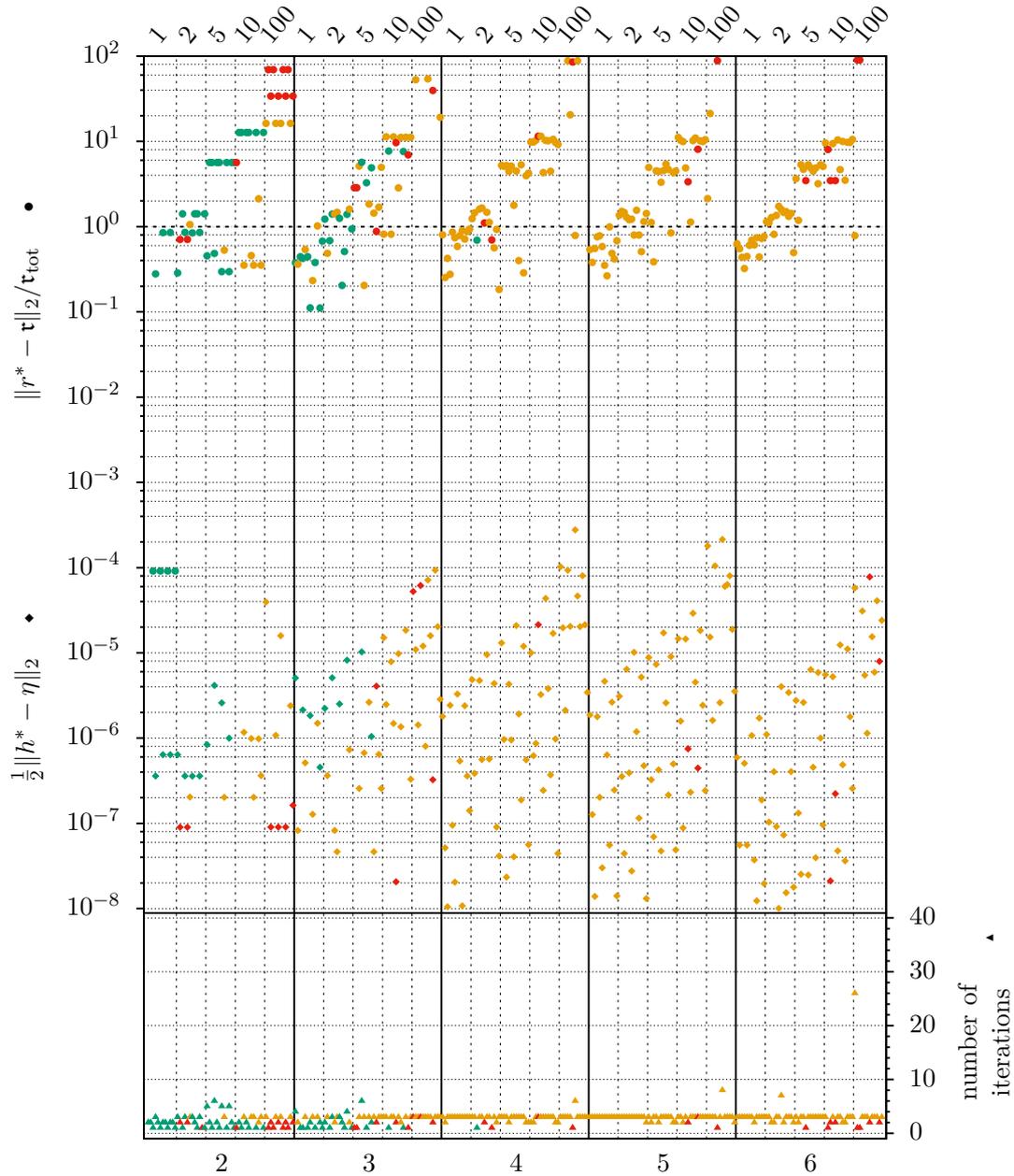


Figure C.2.: Results of all experiments regarding [8000 : uniform : 6 ; 1.0 ; full]. Each of the results is represented by the triple of the number of SQP iterations (tip up triangle) (plotted on the right axis), the final value of the objective (diamond) and the distance of the final iterate to the deployed recombination distribution (bullet) (both plotted on the left axis); the first is separated from the second and third by a solid horizontal line; the bold dotted horizontal line emphasises the value 10^0 . Colouring as described on page 77. An experiment's bullet or diamond is missing if its corresponding value lies outside the fixed plot range $[10^{-8}, 10^2]$. Bottom axis: number of sites (separated by solid vertical lines); top axis: (total) recombination probability used in the simulation scenario (separated by dashed vertical lines). Finer resolution of the order of simulation scenarios as in Figure C.1.

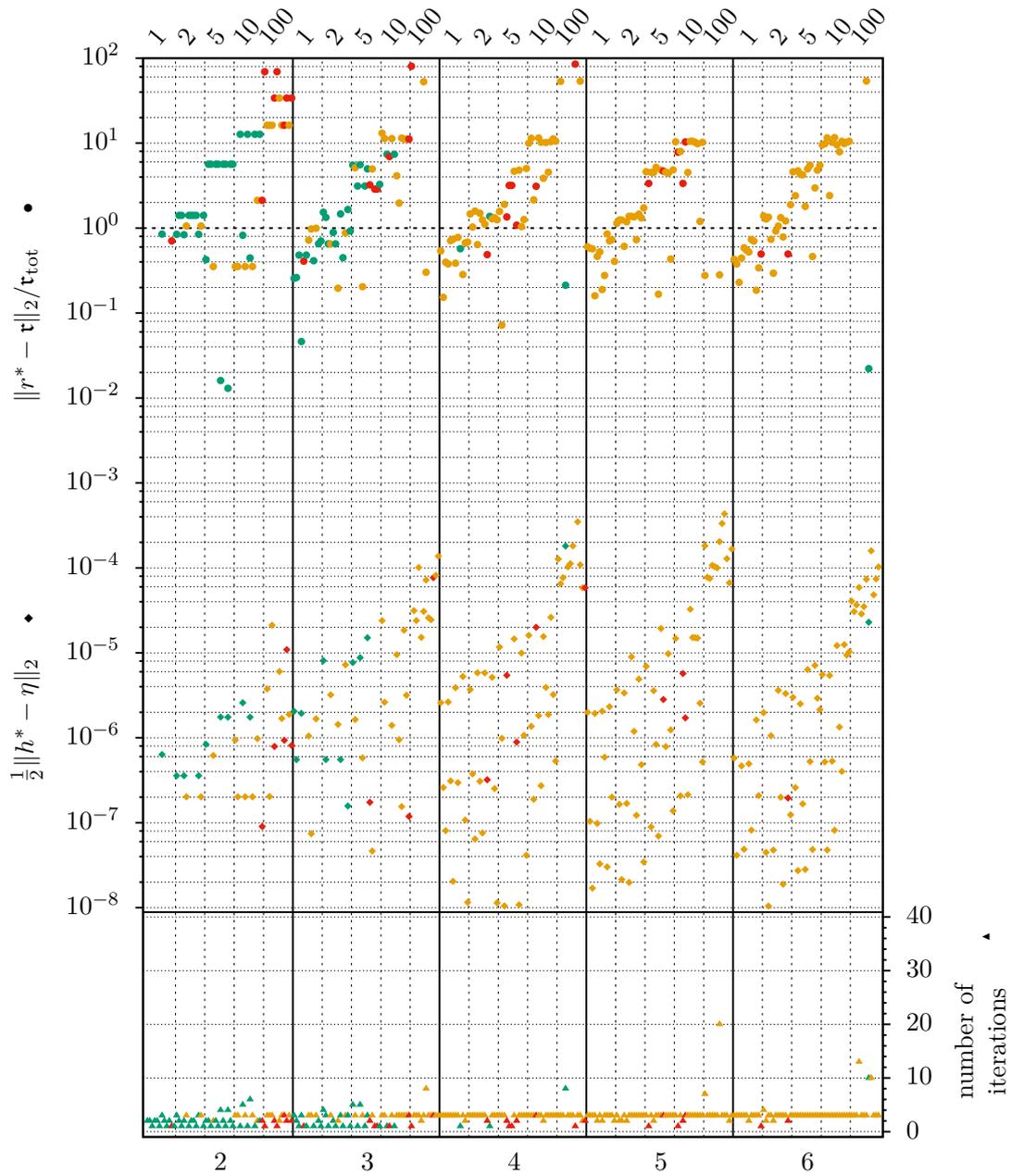


Figure C.3.: Results of all experiments regarding [1600 : uniform : 6 ; 1.0 ; full]. Labels and design as in Figure C.2.

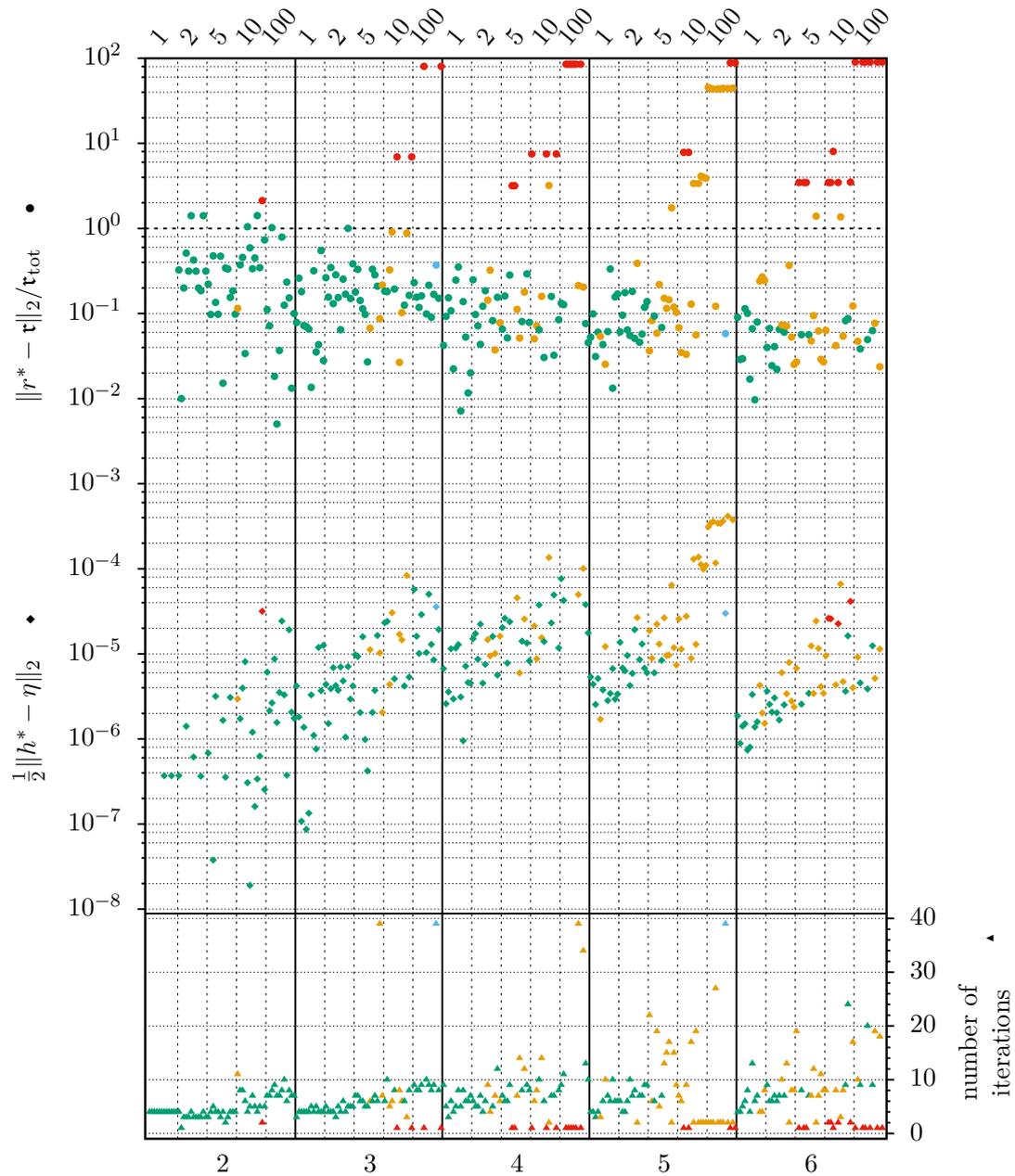


Figure C.4.: Results of all experiments regarding [8000 : logarithmic : 6 ; 1.0 ; full]. Labels and design as in Figure C.2.

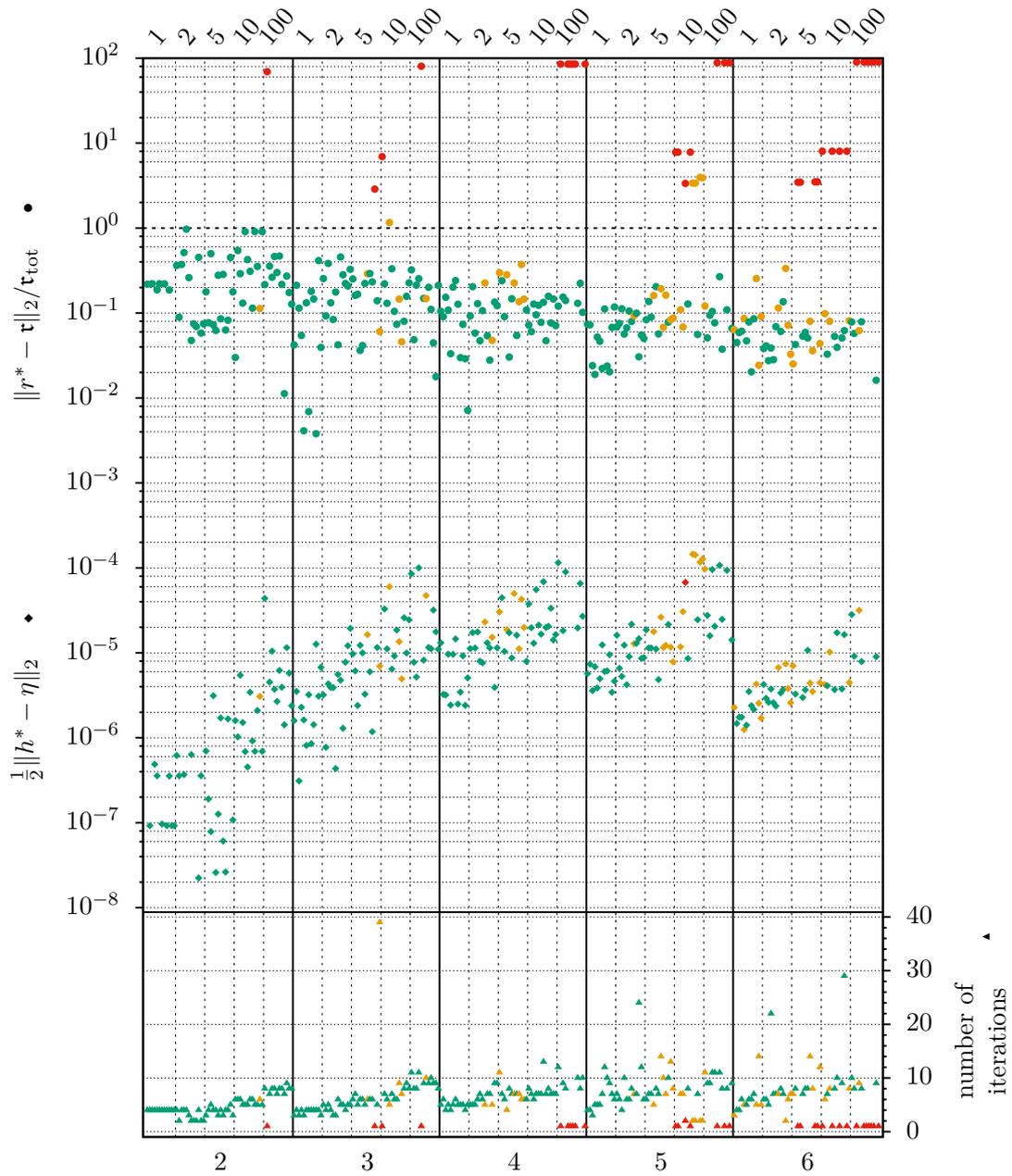


Figure C.5.: Results of all experiments regarding [1600 : logarithmic : 6 ; 1.0 ; full]. Labels and design as in Figure C.2.

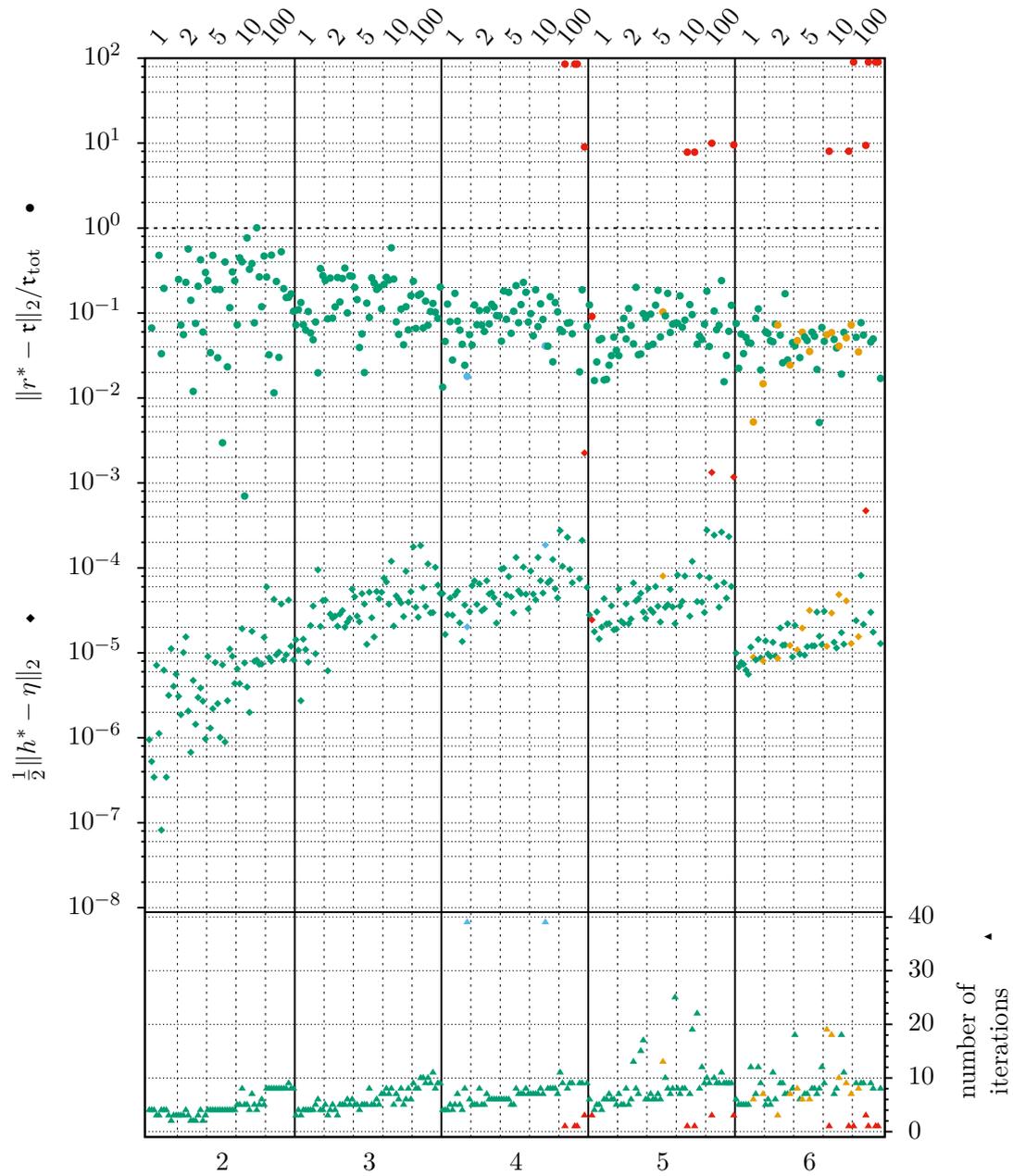


Figure C.6.: Results of all experiments regarding [1600:logarithmic:16;1.0;full]. Labels and design as in Figure C.2.

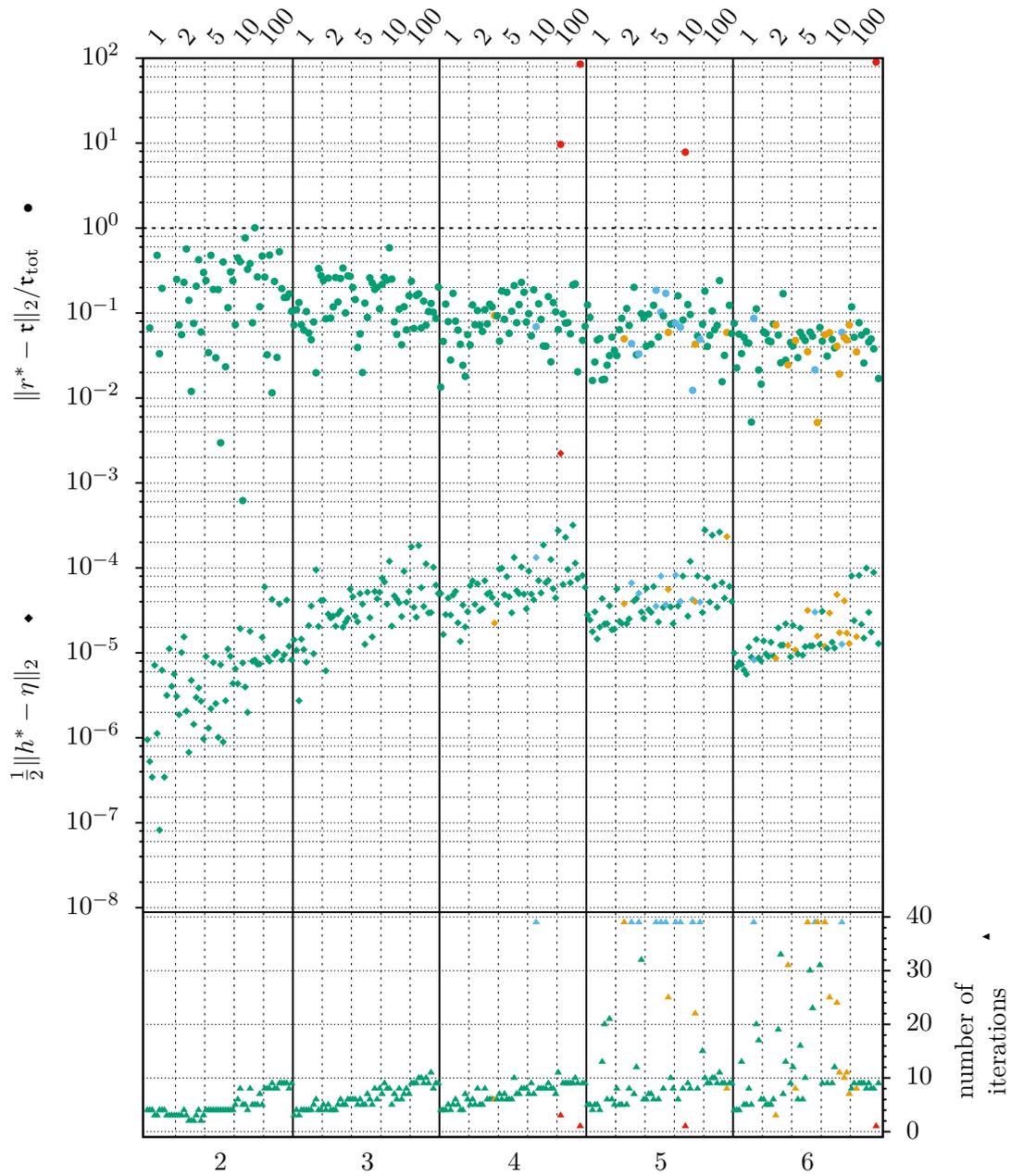


Figure C.7.: Results of all experiments regarding [1600:logarithmic:16;0.1;full]. Labels and design as in Figure C.2.

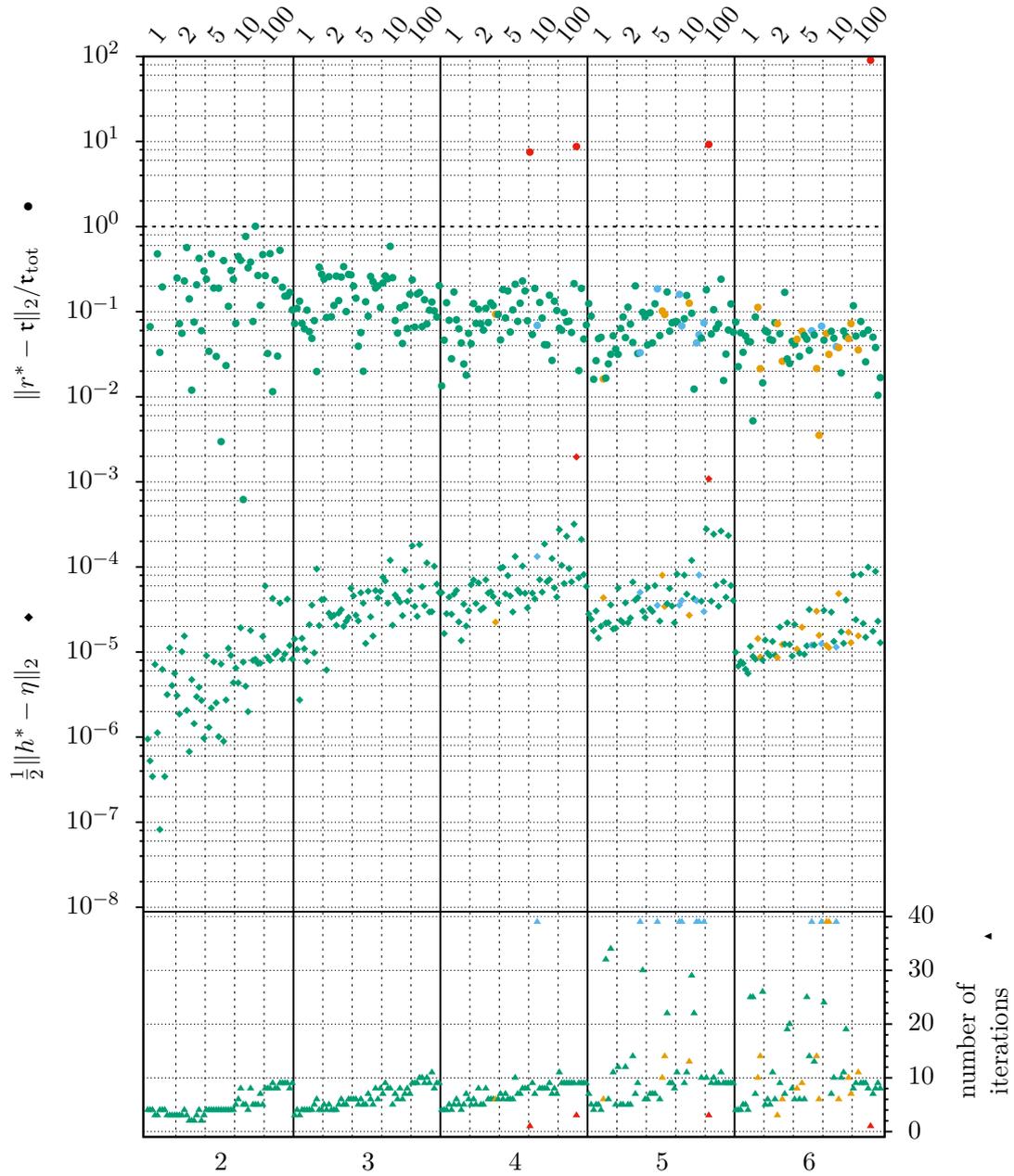


Figure C.8.: Results of all experiments regarding [1600:logarithmic:16;0.1;reduced]. Labels and design as in Figure C.2.

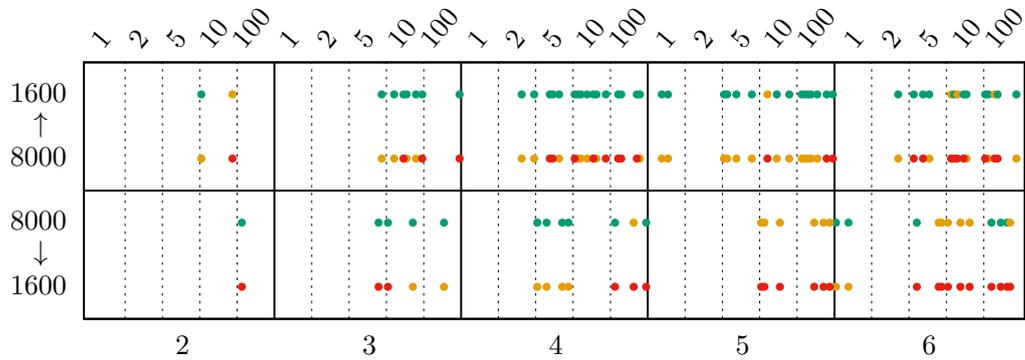


Figure C.9.: Change of exit status by comparing $[8000:\text{logarithmic}:6;1.0;\text{full}]$ with $[1600:\text{logarithmic}:6;1.0;\text{full}]$. Scenarios columnwise as in Figure C.5. Upper part: improvements red \rightarrow orange or green, and orange \rightarrow green; lower part: worsenings green \rightarrow orange or red, and orange \rightarrow red.

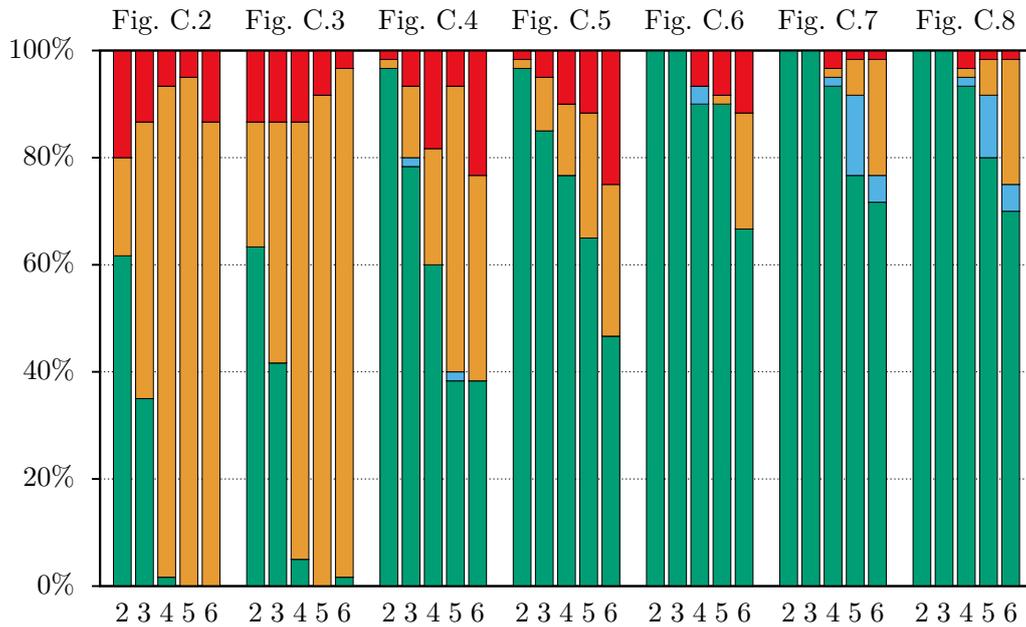


Figure C.10.: Histograms of the experiment's exit status relative frequencies with respect to the number of sites (bottom axis) for each of the considered test cases (referred to by the figure showing its results) (top axis). An increase of the quality from left to right is obvious by this statistics. How close the final iterates r^* are to the deployed τ has to be read from the corresponding results figure.

C.2. Time courses of the averaged normalised sampling functions

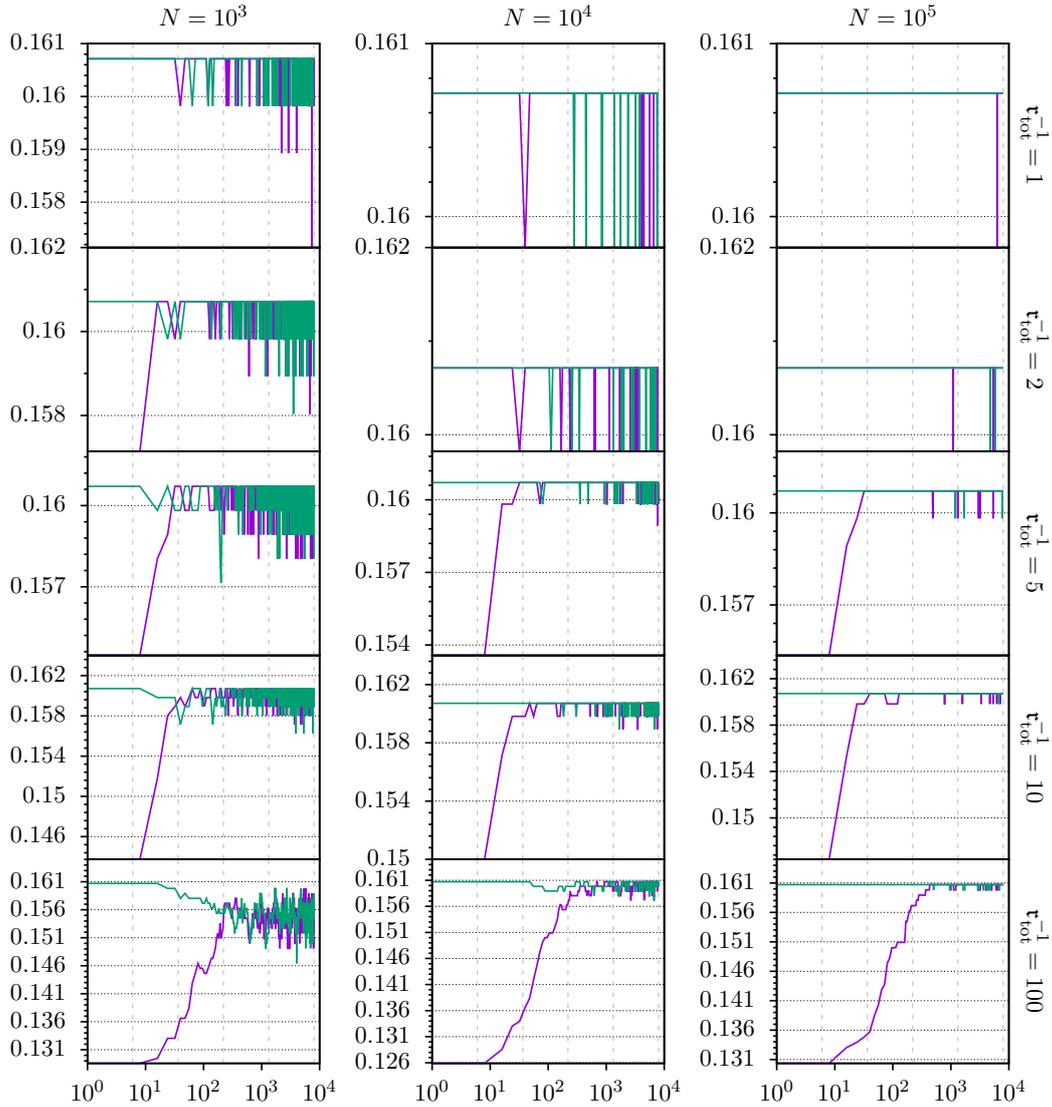


Figure C.11.: Time courses of η for one type ξ , $n = 2$ sites and triangular shape of the recombination distribution. Logarithmic grid with $o = 6$ nodes and time horizon $t_o = 8000$. Plotted on vertical axis against bottom axis. Different colours for the components $\eta_{\mathcal{A}}$, $\mathcal{A} \in \mathbb{P}(S)$.

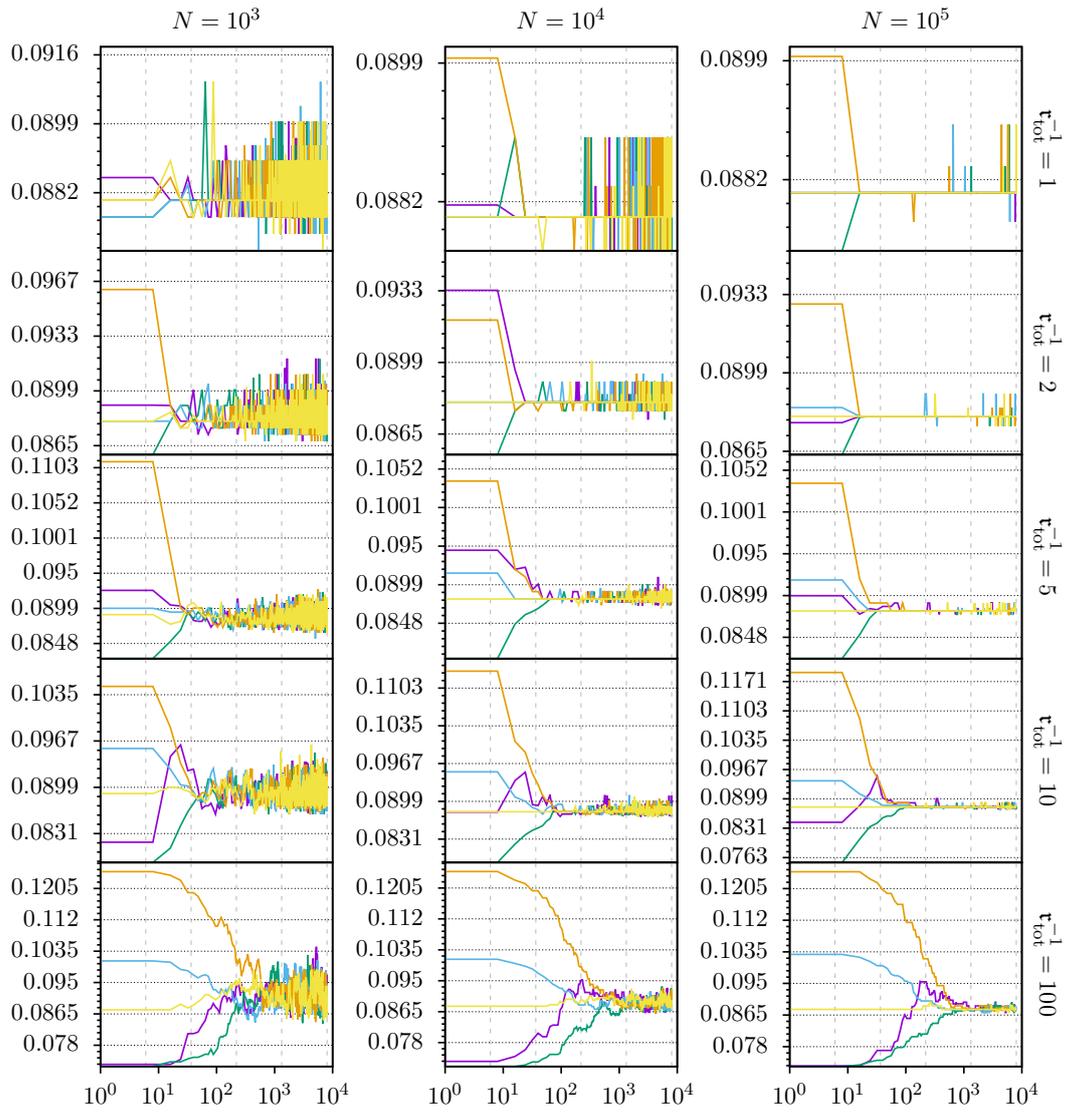


Figure C.12.: Time courses of η for one type ξ and $n = 3$ sites. Further settings and design as in Figure C.11.

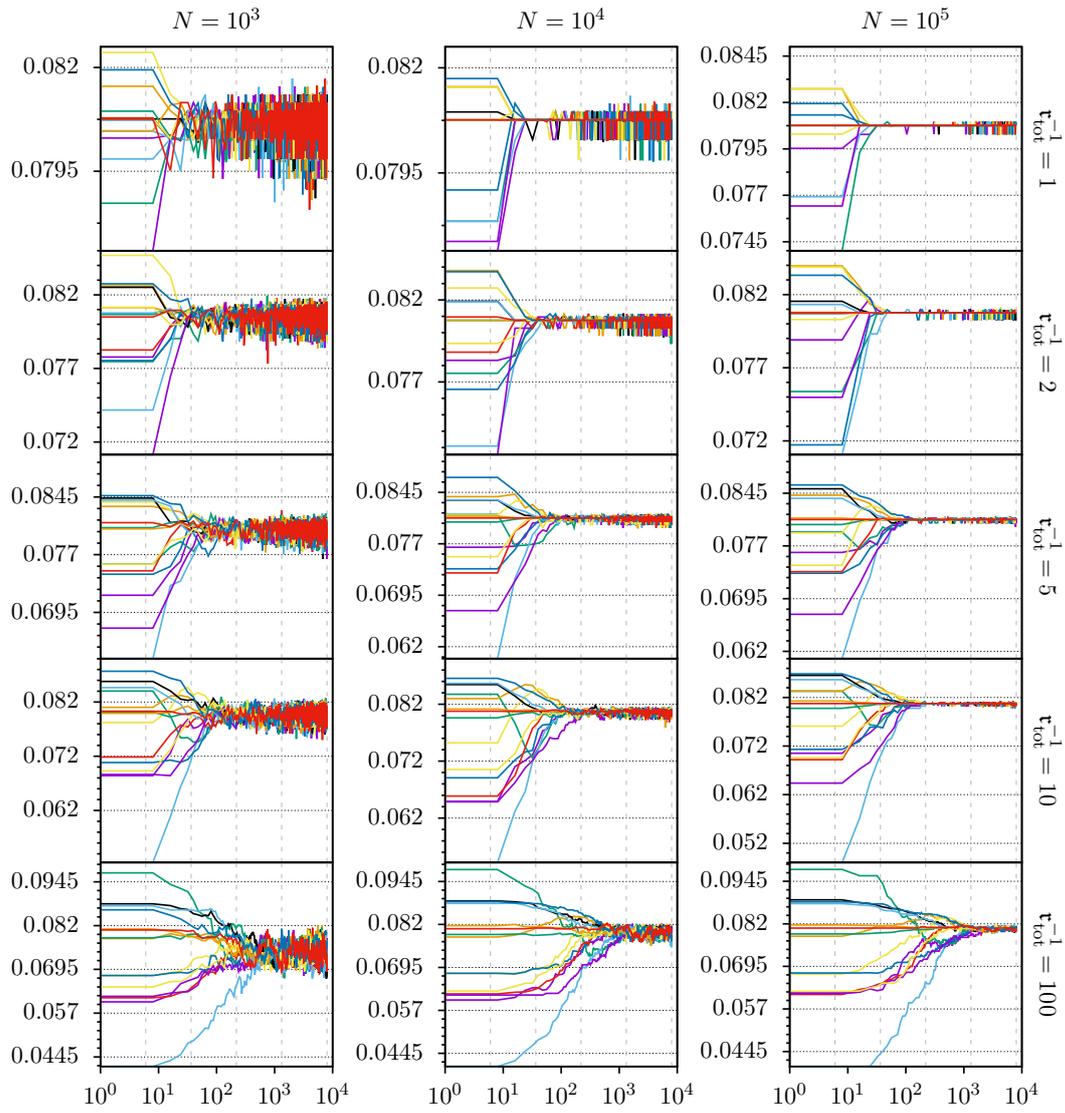


Figure C.13.: Time courses of η for one type ξ and $n = 4$ sites. Further settings and design as in Figure C.11.

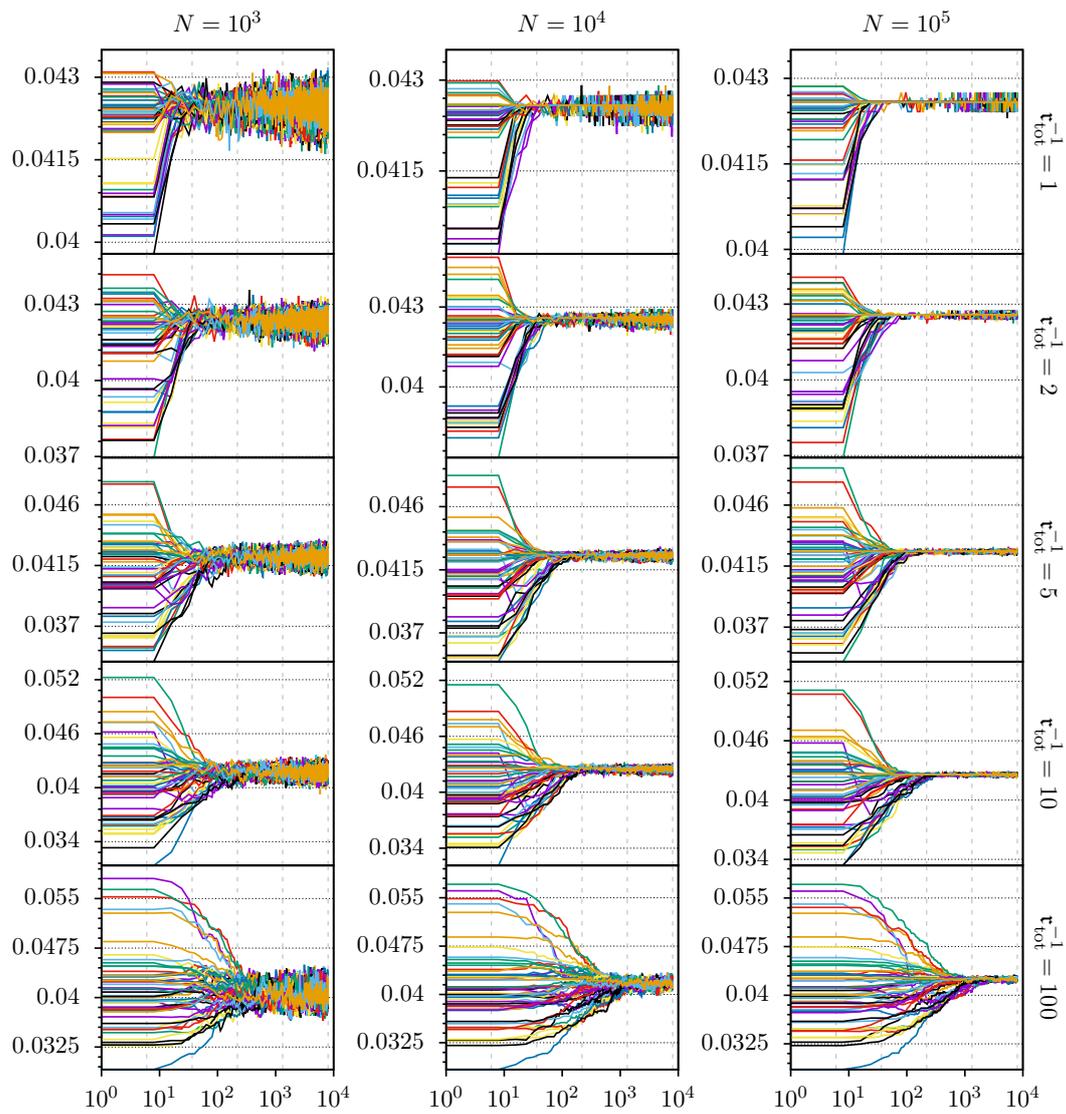


Figure C.14.: Time courses of η for one type ξ and $n = 5$ sites. Further settings and design as in Figure C.11.

C.3. Runtimes of experiments regarding different test cases

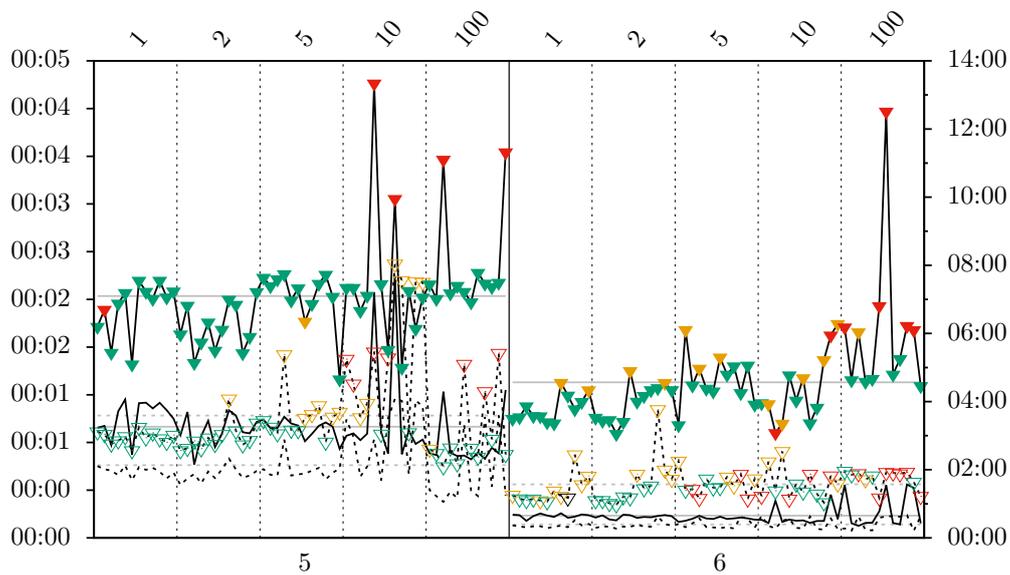


Figure C.15.: Average (per iteration) SQP computation (with tip down triangles) and NLP evaluation (without tip down triangles) CPU times in min regarding [1600:logarithmic:•;1.0;full] with either $o = 16$ (solid) or $o = 6$ (dashed) grid timepoints, restricted to $n = 5, 6$. The connecting lines only serve as visual support to tie corresponding time results of the same test case. Mean values (grey): $n = 5, o = 16$: 00:02.5 min (SQP) and 00:01.2 min (NLP); $n = 6, o = 16$: 04:33.9 min (SQP) and 00:39.1 min (NLP); $n = 5, o = 6$: 00:01.3 min (SQP) and 00:00.8 min (NLP); $n = 6, o = 6$: 01:34.3 min (SQP) and 00:23.8 min (NLP).

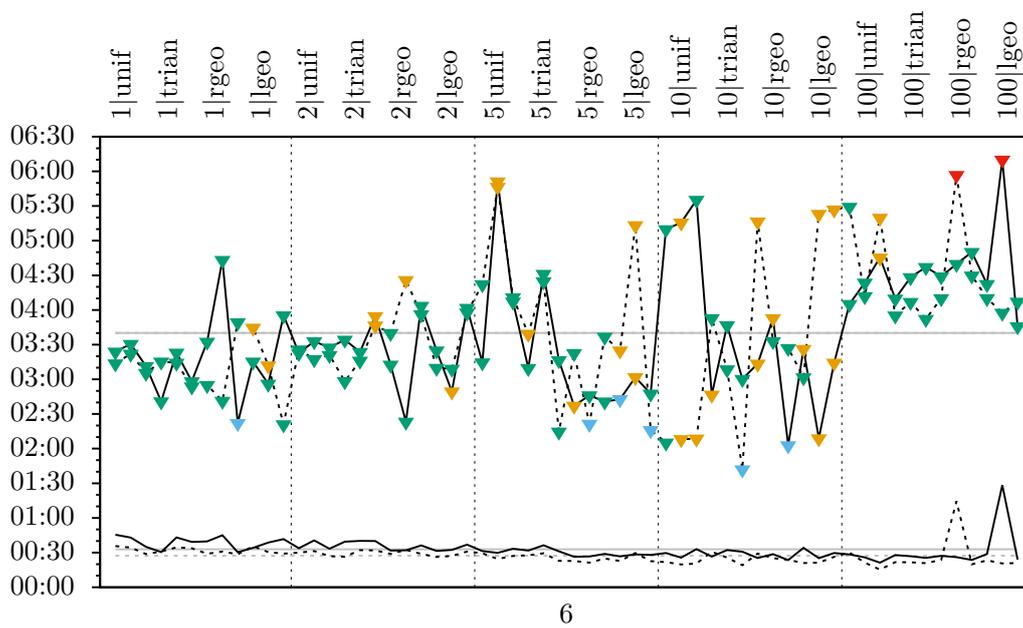


Figure C.16.: Average (per iteration) SQP computation (with tip down triangles) and NLP evaluation (without tip down triangles) CPU times in min regarding $[1600:\text{logarithmic}:16;0.1;\bullet]$ with either full (solid) or reduced (dashed) accuracy, restricted to $n = 6$. The connecting lines only serve as visual support to tie corresponding time results of the same test case. Mean values (grey): full: 03:40.15 min (SQP) and 00:32.84 min (NLP); reduced: 03:40.08 min (SQP) and 00:27.29 min (NLP).

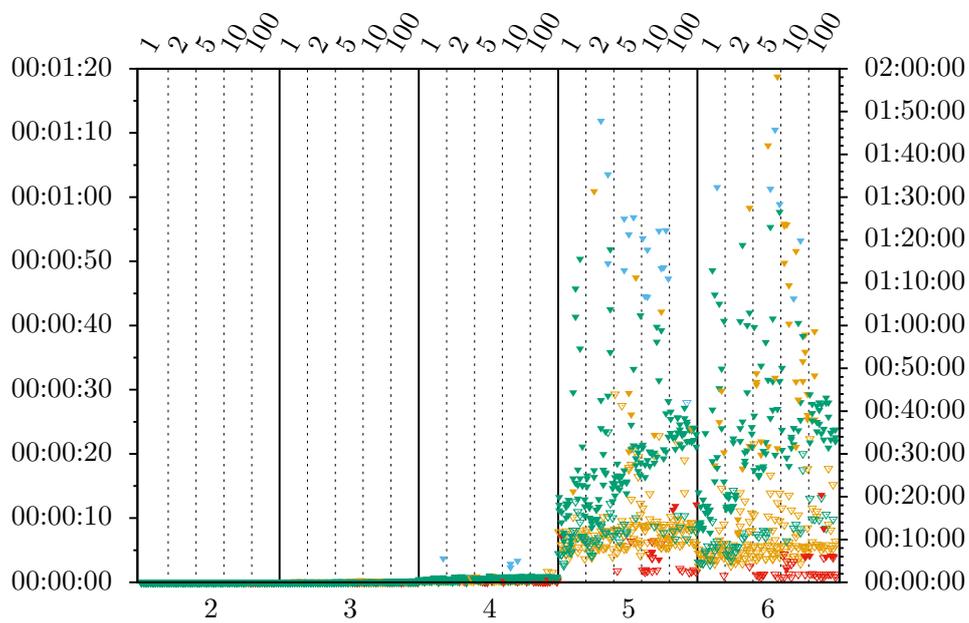


Figure C.17.: Total SQP computation times in h for each experiment (for all test cases). The experiments with $n < 6$ are plotted against the left vertical axes, the ones with $n = 6$ against the right vertical axes. Colouring and ordering of the scenarios as in Figure C.2. Considerable difference between the test cases with $o = 16$ (filled tip down triangle) and $o = 6$ (unfilled tip down triangle).

Bibliography

- [1] M. Aigner. *Combinatorial Theory*. Classics in Mathematics. Springer, Berlin, 1979. Reprint 1997.
- [2] M. Arenas. Computer programs and methodologies for the simulation of DNA sequence data with recombination. *Front. Genet.*, 4:9, 2013.
- [3] E. Baake, M. Baake, and M. Salamat. The general recombination equation in continuous time and its solution. *Discrete Contin. Dyn. Syst.*, 36(1):63–95, 2016.
- [4] E. Baake, A. González Casanova, S. Probst, and A. Wakolbinger. Modelling and simulating Lenski’s long-term evolution experiment. *Submitted*, ArXiv:1803.09995v2, 2018.
- [5] E. Baake and I. Herms. Single-crossover dynamics: Finite versus infinite populations. *Bull. Math. Biol.*, 70(2):603–624, 2008.
- [6] E. Baake and T. Hustedt. Moment closure in a Moran Model with Recombination. *Markov Process. Relat. Fields*, 17(3):429–446, 2011.
- [7] E. Baake and U. von Wangenheim. Single-crossover recombination and ancestral recombination trees. *J. Math. Biol.*, 68(6):1371–1402, 2014.
- [8] M. Baake. Recombination semigroups on measure spaces. *Monatsh. Math.*, 146(4):267–278, 2005.
- [9] M. Baake and E. Baake. An exactly solved model for mutation, recombination and selection. *Canad. J. Math.*, 55(1):3–41, 2003.
- [10] G. Bader and P. Deuffhard. A semi-implicit mid-point rule for stiff systems of ordinary differential equations. *Numer. Math.*, 41(3):373–398, 1983.
- [11] J. E. Barrick, D. S. Yu, S. H. Yoon, H. Jeong, T. K. Oh, D. Schneider, R. E. Lenski, and J. F. Kim. Genome evolution and adaptation in a long-term experiment with *Escherichia coli*. *Nature*, 461:1243–1247, 2009.
- [12] J. Bennett. On the theory of random mating. *Ann. Eugen.*, 18(4):311–317, 1954.
- [13] C. Berge. *Principles of Combinatorics*, volume 72 of *Mathematics in Science and Engineering*. Academic Press, New York, 1971.

-
- [14] A. Bhaskar and Y. S. Song. Closed-form asymptotic sampling distributions under the coalescent with recombination for an arbitrary number of loci. *Adv. Appl. Prob.*, 44(2):391–407, 2012.
- [15] L. Biegler. *Nonlinear Programming: Concepts, Algorithms and Applications to Chemical Processes*. MOS-SIAM Series on Optimization. SIAM, Philadelphia, 2010.
- [16] A. Bobrowski and M. Kimmel. A random evolution related to a Fisher-Wright-Moran model with mutation, recombination and drift. *Math. Methods Appl. Sci.*, 26(18):1587–1599, 2003.
- [17] A. Bobrowski, T. Wojdyla, and M. Kimmel. Asymptotic behavior of a Moran model with mutations, drift and recombination among multiple loci. *J. Math. Biol.*, 61(3):455–473, 2010.
- [18] H. G. Bock. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*. Ph. D. dissertation, Bonner Mathematische Schriften 183, Universität Bonn, 1987.
- [19] H. G. Bock, E. Kostina, and J. P. Schlöder. Direct multiple shooting and generalized Gauss-Newton method for parameter estimation problems in ODE models. In T. Carraro, M. Geiger, S. Körkel, and R. Rannacher, editors, *Multiple Shooting and Time Domain Decomposition Methods*, volume 9 of *Contributions in Mathematical and Computational Sciences*, pages 1–34. Springer, Cham, 2015.
- [20] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004.
- [21] R. Bürger. *The mathematical Theory of Selection, Recombination, and Mutation*. Wiley Series in Mathematical and Computational Biology. John Wiley & Sons, Chichester, 2000.
- [22] I. Cassens, P. Mardulyn, and M. M. C. Evaluating intraspecific “network” construction methods using simulated sequence data: Do existing algorithms outperform the global maximum parsimony approach? *Syst. Biol.*, 54(3):363–372, 2005.
- [23] L.-M. Chevin. On measuring selection in experimental evolution. *Biol. Lett.*, 7:210–213, 2011.
- [24] M. M. Desai and D. S. Fisher. Beneficial mutation–selection balance and the effect of linkage on positive selection. *Genetics*, 176(3):1759–1798, 2007.
- [25] P. Donnelly. Dual processes in population genetics. In P. Tautu, editor, *Stochastic Spatial Processes*, Lecture Notes in Mathematics, pages 94–105. Springer, Berlin, 1986.

-
- [26] R. Durrett. *Probability models for DNA Sequence Evolution*. Probability and Its Applications. Springer, New York, 2nd edition, 2008.
- [27] R. Durrett and J. Mayberry. Traveling waves of selective sweeps. *Ann. Appl. Probab.*, 21(2):699–744, 2011.
- [28] F. J. Dyson. Statistical theory of the energy levels of complex systems. III. *J. Math. Phys.*, 3:166–175, 1962.
- [29] M. Esser, S. Probst, and E. Baake. Partitioning, duality, and linkage disequilibria in the moran model with recombination. *J. Math. Biol.*, 73(1):161–197, 2016.
- [30] S. N. Ethier and T. G. Kurtz. *Markov Processes: Characterization and Convergence*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, Chichester, 1986. Reprint 2005.
- [31] W. J. Ewens. *Mathematical Population Genetics*. Interdisciplinary Applied Mathematics. Springer, New York, 2nd edition, 2004.
- [32] W. J. Ewens and G. Thomson. Properties of equilibria in multi-locus genetic systems. *Genetics*, 87(4):807–819, 1977.
- [33] A. Eyre-Walker and P. D. Keightley. The distribution of fitness effects of new mutations. *Nat. Rev. Genet.*, 8:610–618, 2007.
- [34] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume I. John Wiley & Sons, New York, 3rd edition, 1968.
- [35] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume II. John Wiley & Sons, New York, 2nd edition, 1971.
- [36] R. Fisher. The correlation between relatives on the supposition of mendelian inheritance. *Philos. Trans. R. Soc. Edinburgh*, 52:399–433, 1918.
- [37] R. Fletcher. *Practical Methods of Optimization*. A Wiley-Interscience Publication. John Wiley & Sons, Chichester, 2nd edition, 1987.
- [38] C. A. Fogle, J. L. Nagle, and M. M. Desai. Clonal interference, multiple mutations and adaptation in large asexual populations. *Genetics*, 180(4):2163–2173, 2008.
- [39] H. Geiringer. On the probability theory of linkage in Mendelian heredity. *Ann. Math. Statist.*, 15(1):25–57, 1944.
- [40] H. O. Gerogii. *Stochastics*. De Gruyter Textbook. De Gruyter, Berlin, 2nd edition, 2013.
- [41] P. J. Gerrish. The rhythm of microbial adaptation. *Nature*, 413:299–302, 2001.

- [42] P. J. Gerrish and R. E. Lenski. The fate of competing beneficial mutations in an asexual population. *Genetica*, 102/103:127–144, 1998.
- [43] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization*. Addison Wesley, 1981.
- [44] P. E. Gill and E. Wong. Sequential quadratic programming methods. In J. Lee and S. Leyffer, editors, *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 147–224. Springer, New York, 2012.
- [45] J. H. Gillespie. Molecular evolution over the mutational landscape. *Evolution*, 38(5):1116–1129, 1984.
- [46] G. B. Golding. The sampling distribution of linkage disequilibrium. *Genetics*, 108(1):257–274, 1984.
- [47] A. González Casanova, N. Kurt, A. Wakolbinger, and L. Yuan. An individual-based model for the Lenski experiment, and the deceleration of the relative fitness. *Stoch. Proc. Appl.*, 126(8):2211–2252, 2016.
- [48] B. H. Good and M. M. Desai. The impact of macroscopic epistasis on long-term evolutionary dynamics. *Genetics*, 199(1):177–190, 2015.
- [49] B. H. Good, M. J. McDonald, J. E. Barrick, R. E. Lenski, and M. M. Desai. The dynamics of molecular evolution over 60,000 generations. *Nature*, 551:45–50, 2017.
- [50] R. Gorelick and M. D. Laubichler. Decomposing multilocus linkage disequilibrium. *Genetics*, 166(3):1581–1583, 2004.
- [51] W. B. Gragg. On extrapolation algorithms for ordinary initial value problems. *J. SIAM Numer. Anal. Ser.B*, 2(3):384–403, 1965.
- [52] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete mathematics: A Foundation for Computer Science*. Addison-Wesley, Upper Saddle River, NJ, 2 edition, 1994.
- [53] D. Graur and W.-H. Li. *Fundamentals of Molecular Evolution*. Sinauer, Sunderland, MA, 2nd edition, 2000.
- [54] R. C. Griffiths and P. Marjoram. Ancestral inference from samples of DNA sequences with recombination. *J. Comp. Biol.*, 3(4):479–502, 1996.
- [55] A. Hastings. Linkage disequilibrium, selection and recombination at three loci. *Genetics*, 106(1):153–164, 1984.
- [56] J. Hein, M. H. Schierup, and C. Wiuf. *Gene genealogies, variation and evolution: A Primer in Coalescent Theory*. Oxford University Press, Oxford, 2005.

- [57] R. R. Hudson. Properties of a neutral allele model with intragenetic recombination. *Theor. Popul. Biol.*, 23(2):183–201, 1983.
- [58] R. R. Hudson. Generating samples under a Wright-Fisher neutral model of genetic variation. *Bioinformatics*, 18(2):337–338, 2002.
- [59] S. Jansen and N. Kurt. On the notion(s) of duality for Markov processes. *Prob. Surveys*, 11:59–120, 2014.
- [60] P. A. Jenkins, P. Fearnhead, and Y. S. Song. Tractable diffusion and coalescent processes for weakly correlated loci. *Electron. J. Probab.*, 20(58):1–26, 2015.
- [61] P. A. Jenkins and R. C. Griffiths. Inference from samples of DNA sequences using a two-locus model. *J. Comp. Biol.*, 18(1):109–127, 2011.
- [62] P. A. Jenkins and Y. S. Song. An asymptotic sampling formula for the coalescent with recombination. *Ann. Appl. Probab.*, 20(3):1005–1028, 2010.
- [63] J. G. Kemeny and J. L. Snell. *Finite Markov Chains*. Undergraduate Texts in Mathematics. Springer, New York, 1960. Reprint 1981.
- [64] J. S. LeClair and L. M. Wahl. The impact of population bottlenecks on microbial adaptation. *J. Stat. Phys.*, 172(1):114–125, 2018.
- [65] R. E. Lenski, M. R. Rose, S. C. Simpson, and S. C. Tadler. Long-term experimental evolution in *Escherichia coli*. I. Adaptation and divergence during 2,000 generations. *Am. Nat.*, 138(6):1315–1341, 1991.
- [66] R. E. Lenski and M. Travisano. Dynamics of adaptation and diversification: a 10,000-generation experiment with bacterial populations. *Proc. Natl. Acad. Sci. U.S.A.*, 91(15):6808–6814, 1994.
- [67] T. M. Liggett. *Interacting Particle Systems*. Classics in Mathematics. Springer, Berlin, 1981. Reprint 2005.
- [68] S. Mano. Duality between the two-locus Wright-Fisher diffusion model and the ancestral process with recombination. *J. Appl. Prob.*, 50(1):256–271, 2013.
- [69] J. Masel. Genetic drift. *Curr. Biol.*, 21(20):R837–R838, 2011.
- [70] G. A. T. McVean and N. J. Cardin. Approximating the coalescent with recombination. *Philos. Trans. R. Soc. Lond. B. Biol. Sci.*, 360(1459):1387–1393, 2005.
- [71] M. L. Mehta. *Random Matrices*. Academic Press, San Diego, 1991.
- [72] M. Möhle. Forward and backward diffusion approximations for haploid exchangeable population models. *Stoch. Proc. Appl.*, 95(1):133–149, 2001.

- [73] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, Berlin, 2nd edition, 2006.
- [74] T. Ohta and M. Kimura. Linkage disequilibrium due to random genetic drift. *Genet. Res.*, 13(1):47–55, 1969.
- [75] H. A. Orr. The distribution of fitness effects among beneficial mutations. *Genetics*, 163(4):1519–1526, 2003.
- [76] S.-C. Park and J. Krug. Clonal interference in large populations. *Proc. Natl. Acad. Sci. U.S.A.*, 104(46):18135–18140, 2007.
- [77] Z. Patwa and L. M. Wahl. The fixation probability of beneficial mutations. *J. Royal Soc. Interface*, 5(28):1279–1289, 2008.
- [78] P. Phillips, S. Otto, and M. C. Whitlock. Beyond the average: the evolutionary importance of gene interactions and variability of epistatic effects. In J. B. Wolf, E. D. Brodie, and M. J. Wade, editors, *Epistasis and the Evolutionary Process*, pages 20–38. Oxford University Press, Oxford, 2000.
- [79] J. Polańska and M. Kimmel. A model of dynamics of mutation, genetic drift and recombination in DNA-repeat genetic loci. *Arch. Control. Sci.*, 9(1-2):143–157, 1999.
- [80] J. Polańska and M. Kimmel. A simple model of linkage disequilibrium and genetic drift in human genomic SNPs: Importance of demography and SNP age. *Hum. Hered.*, 60(4):181–195, 2005.
- [81] M. D. Rasmussen, M. J. Hubisz, I. Gronau, and A. Siepel. Genome-wide inference of ancestral recombination graphs. *PLoS Genet*, 10(5):e1004342, 2014.
- [82] D. Rose. *Warm Started Active Set Solver for Tree-structured QPs*. Ph. D. dissertation, Leibniz Universität Hannover, 2017.
- [83] G.-C. Rota. On the foundations of combinatorial theory I. Theory of Möbius functions. *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, 2(4):340–368, 1964.
- [84] D. E. Rozen, J. A. G. M. de Visser, and P. J. Gerrish. Fitness effects of fixed beneficial mutations in microbial populations. *Curr. Biol.*, 12(12):1040–1045, 2002.
- [85] R. Sanjuán. Mutational fitness effects in RNA and single-stranded DNA viruses: common patterns revealed by site-directed mutagenesis studies. *Philos. Trans. R. Soc. Lond. B. Biol. Sci.*, 365(1548):1975–1982, 2010.
- [86] Y. S. Song and J. S. Song. Analytic computation of the expectation of the linkage disequilibrium coefficient r^2 . *Theor. Popul. Biol.*, 71(1):49–60, 2007.

-
- [87] R. P. Stanley. *Enumerative Combinatorics*, volume 1 of *The Wadsworth & Brooks/Cole Mathematics Series*. Wadsworth & Brooks/Cole, Monterey, CA, 1986.
- [88] A. Stuart and K. Ord. *Distribution Theory*, volume 1 of *Kendall's Advanced Theory of Statistics*. John Wiley & Sons, Chichester, 5. edition, 1994.
- [89] O. Tenaillon, J. E. Barrick, N. Ribeck, D. E. Deatherage, J. L. Blanchard, A. Dasgupta, G. C. Wu, S. Wielgoss, S. Cruveiller, C. Médigue, D. Schneider, and R. E. Lenski. Tempo and mode of genome evolution in a 50,000-generation experiment. *Nature*, 536(7615):165–170, 2016.
- [90] U. von Wangenheim, E. Baake, and M. Baake. Single-crossover recombination in discrete time. *J. Math. Biol.*, 60(5):727–760, 2010.
- [91] A. Wächter and L. T. Biegler. Line search filter methods for nonlinear programming: Local convergence. *SIAM J. Optim.*, 16(1):32–48, 2005.
- [92] A. Wächter and L. T. Biegler. Line search filter methods for nonlinear programming: Motivation and global convergence. *SIAM J. Optim.*, 16(1):1–31, 2005.
- [93] L. M. Wahl and A. D. Zhu. Survival probability of beneficial mutations in bacterial batch culture. *Genetics*, 200(1):309–320, 2015.
- [94] J. Wakeley. *Coalescent Theory: An Introduction*. Roberts and Co., Greenwood Village, CO, 2009.
- [95] Y. Wang and B. Rannala. Bayesian inference of fine-scale recombination rates using population genomic data. *Philos. Trans. R. Soc. Lond. B. Biol. Sci.*, 363(1512):3921–3930, 2008.
- [96] M. J. Wisner, N. Ribeck, and R. E. Lenski. Data from: Long-term dynamics of adaptation in asexual populations. *Science*, 2013.
- [97] M. J. Wisner, N. Ribeck, and R. E. Lenski. Long-term dynamics of adaptation in asexual populations. *Science*, 342(6164):1364–1367, 2013.
- [98] C. Wiuf and J. Hein. On the number of ancestors to a DNA sequence. *Genetics*, 147(3):1459–1468, 1997.
- [99] S. M. Woolley, D. Posada, and K. A. Crandall. A comparison of phylogenetic network methods using computer simulation. *PLoS ONE*, 3(4):e1913, 2008.
- [100] A. Wünsche, D. M. Dinh, R. S. Satterwhite, C. D. Arenas, D. M. Stoebel, and T. F. Cooper. Diminishing-returns epistasis decreases adaptability along an evolutionary trajectory. *Nat. Ecol. Evol.*, 1(4):0061, 2017.