Stochastic spatial modelling of DNA methylation patterns and moment-based parameter estimation

Alexander Tobias Lück

Dissertation

zur Erlangung des Grades
des Doktors der Naturwissenschaften (Dr. rer. nat.)
der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

Saarbrücken, Deutschland
Juni 2020

Dean of the Faculty	Prof. Dr. Thomas Schuster
Day of Colloquium	11. September 2020

Examination Board:

Chairman	Prof. Dr. Volkhard Helms
Reviewers	Prof. Dr. Verena Wolf
	Prof. Dr. Luca Bortolussi
Academic Assistant	Dr. Felix Martin Schuhknecht

Acknowledgements

At this point, I would to thank all the people who have supported me.

First and foremost, I would like to thank my supervisor Prof. Dr. Verena Wolf for giving me, a (former) physicist, the opportunity to take up a new challenge in persuing a PhD in computer science and for introducing me to the exciting field of epigentics.

Also a huge thank you to my (former) colleagues Michael Backenköhler, Timo P. Gros, Gerrit Großmann, Charalampos Kyriakopoulos, and Thilo Krüger. It was a pleasure to work with you. Thanks to Pascal Giehr and Jörn Walter from the epigenetics department for the fruitful discussions, for their biological insights and for the great collaboration.

Next, I would like to thank the people from outside the university for providing a welcome change. First, I would like to mention my friends Anne Hafner, Johannes Geiser, Michael Schweitzer, and Sidney Tregellis here, for always willing to listen to my problems, for giving moral support and for the fun we have in general. Furthermore, I would like to thank my comrades from the "Freiwillige Feuerwehr Bierbach", the athletes and coaches from my track and field team "LG Bliestal", and my gym buddies.

Last but certainly not least I would like to express my deepest gratidude to my family, especially my parents Stephanie and Lorenz, as well as my brothers Julian and Robin. Without your tireless support in every possible way, I would not be the person I am today. Thank you for everything!

Abstract

In the first part of this thesis, we introduce and analyze spatial stochastic models for DNA methylation, an epigenetic mark with an important role in development. The underlying mechanisms controlling methylation are only partly understood. Several mechanistic models of enzyme activities responsible for methylation have been proposed. Here, we extend existing hidden Markov models (HMMs) for DNA methylation by describing the occurrence of spatial methylation patterns with stochastic automata networks. We perform numerical analysis of the HMMs applied to (non-)hairpin bisulfite sequencing KO data and accurately predict the wild-type data from these results. We find evidence that the activities of Dnmt3a/b responsible for *de novo* methylation depend on the left but not on the right CpG neighbors.

The second part focuses on parameter estimation in chemical reaction networks (CRNs). We propose a generalized method of moments (GMM) approach for inferring the parameters of CRNs based on a sophisticated matching of the statistical moments of the stochastic model and the sample moments of population snapshot data. The proposed parameter estimation method exploits recently developed moment-based approximations and provides estimators with desirable statistical properties when many samples are available. The GMM provides accurate and fast estimations of unknown parameters of CRNs. The accuracy increases and the variance decreases when higher-order moments are considered.

Zusammenfassung

Im ersten Teil der Arbeit führen wir eine Analyse für spatielle stochastische Modelle der DNA Methylierung, ein wichtiger epigenetischer Marker in der Entwicklung, durch. Die zugrunde liegenden Mechanismen der Methylierung werden noch nicht vollständig verstanden. Mechanistische Modelle beschreiben die Aktivität der Methylierungsenzyme. Wir erweitern bestehende Hidden Markov Models (HMMs) zur DNA Methylierung durch eine Stochastic Automata Networks Beschreibung von spatiellen Methylierungsmustern. Wir führen eine numerische Analyse der HMMs auf bisulfit-sequenzierten KO Datensätzen aus und nutzen die Resultate, um die Wildtyp-Daten erfolgreich vorherzusagen. Unsere Ergebnisse deuten an, dass die Aktivitäten von Dnmt3a/b, die überwiegend für die *de novo* Methylierung verantwortlich sind, nur vom Methylierungsstatus des linken, nicht aber vom rechten CpG Nachbarn abhängen.

Der zweite Teil befasst sich mit Parameterschätzung in chemischen Reaktionsnetzwerken (CRNs). Wir führen eine Verallgemeinerte Momentenmethode (GMM) ein, die die statistischen Momente des stochastischen Modells an die Momente von Stichproben geschickt anpasst. Die GMM nutzt hier kürzlich entwickelte, momentenbasierte Näherungen, liefert Schätzer mit wünschenswerten statistischen Eigenschaften, wenn genügend Stichproben verfügbar sind, mit schnellen und genauen Schätzungen der unbekannten Parameter in CRNs. Momente höherer Ordnung steigern die Genauigkeit des Schätzers, während die Varianz sinkt.

Publications

This thesis is based on the following publications. For each publication the corresponding parts of this thesis are also listed below:

- A. Lück, V. Wolf:
  Generalized method of moments for estimating parameters of stochastic reaction networks
  BMC Systems Biology (2016) [88] (Chapter 3 and Section 2.4.1)

- A. Lück, P. Giehr, J. Walter, V. Wolf:
  A Stochastic Model for the Formation of Spatial Methylation Patterns
  International Conference on Computational Methods in Systems Biology (2017) [87] (Sections 2.3.1, 2.3.2, 2.3.3, 2.3.4, 2.3.6, 2.4.3, 2.5.1, 2.5.3 and Appendix A)

- A. Lück, P. Giehr, K. Nordström, J. Walter, V. Wolf:
  Hidden Markov Modelling Reveals Neighborhood Dependence of Dnmt3a and 3b Activity
  IEEE/ACM Transactions on Computational Biology and Bioinformatics (2019) [86] (Sections 2.3.1, 2.3.2, 2.3.3, 2.3.4, 2.3.6, 2.3.7, 2.4.3, 2.5)

- A. Lück, V. Wolf:
  Generalized Method of Moments Estimation for Stochastic Models of DNA Methylation Patterns
  submitted for publication (2019) [89] (Sections 2.4.1, 2.5.1)

- A. Lück, V. Wolf:
  A Stochastic Automata Network Description for Spatial DNA-Methylation Models
  International Conference on Measurement, Modelling and Evaluation of Computing Systems (2020) [90] (Section 2.3.5)

All content (text passages, figures, tables) from these publications that appears in this thesis is originally the author's work. Exceptions are clearly marked and acknowledged.

The following publications are not covered in this thesis. The topics of these publications are outside of the scope of the material covered here.

- C. Arita, A. Lück, L. Santen:
  Length regulation of microtubules by molecular motors: exact solution and density profiles
  Journal of Statistical Mechanics: Theory and Experiment (2015) [8]

- A. Lück:
  Replicated Computational Results (RCR) Report for "Automatic Moment-Closure Approximation of Spatially Distributed Collective Adaptive Systems"
  ACM TOMACS (2016) [85]

- C. Arita, J. Bosche, A. Lück, L. Santen:
  Localization of a microtubule organizing center by kinesin motors
  Journal of Statistical Mechanics: Theory and Experiment (2017) [7]

- P. Kurasov, A. Lück, D. Mugnolo, V. Wolf:
  Stochastic hybrid models of gene regulatory networks-A PDE approach
  Mathematical Biosciences (2018) [74]

- C. Kyriakopoulos, P. Giehr, A. Lück, J. Walter, V. Wolf:
  A Hybrid HMM Approach for the Dynamics of DNA Methylation
  Workshop on Hybrid Systems & Biology (2019) [77]

Contents

Chapter 1

Introduction

This thesis consists of two parts, which are (mostly) independent of each other. Therefore, each of the two chapters is self-contained with its own extensive introduction, such that we keep this general introduction rather short. Nevertheless, we give a brief outlook on both chapters here and summarize the main contributions of this thesis.

In Chapter 2 we introduce a generalization of existing hidden Markov models for DNA methylation, which can model whole sequences of CpGs, a special combination of base pairs in the DNA, important for methylation, at once. To this end, we introduce dependence parameters and appropriate transition probability functions for different methylation states of the neighborhood, such that we can take correlations and dependencies between adjacent CpGs into account. Compared to single CpG models, our model is much more complex and has higher requirements on the biological data. The generation of the transition probability matrices for our model can be formalized by a stochastic automata network description, given the matrices for a single CpG model and suitable transition probability functions that depend on the neighborhood. We check the applicability of different parameter estimation methods (amongst others the generalized method of moments from Chapter 3, modified for moments of methylation patterns, which will later be used in the context of chemical reaction networks) and apply our model to biological data from selected loci within the genome, as well as whole genome data.

In Chapter 3 we apply the generalized method of moments (GMM), a well-known parameter estimation technique from econometrics in a biological context, more specifically for chemical reaction networks. We test different methods to obtain a weight matrix and investigate the influence higher order moments. Since, in general, not all reactions are monomolecular, the exact moments are not obtainable and we have to rely on approximations (moment closure). We

also investigate the influence of the quality of the approximated moments on the estimation quality by using standard and hybrid closure techniques.

The main contributions of this thesis are:

- The introduction of a generalized hidden Markov model to model multiple CpGs at once, which enables the investigation of the influence of the methylation states from adjacent CpGs and to test different hypotheses about the working mechanisms of the methylation enzymes (Dnmts).
- A stochastic automata networks description that allows to generalize every single CpG model to multiple CpGs, given the transition matrix and a suitable neighborhood function.
- New biological insights and confirmation of existing hypotheses due to our mechanistic model: Dnmt1 works processively and independent of the neighborhood, while Dnmt3a/b shows a dependence to the left. Hypomethylated CpGs im promoter regions tend to behave more independently, while hypermethylated CpGs from other regions show a stronger dependence only to the left.
- The introduction of the well-known GMM estimator from econometrics to biological applications, especially chemical reaction networks and methylation patterns.
- The investigation of the influence on the estimation quality of the GMM, if approximated moments are used.

Chapter 2

Stochastic Modelling of Spatial Methylation Patterns

2.1 Introduction

The DNA contains the blueprints for all proteins that can be expressed within an organism and thus determines its appearance and behavior. However, the differences in the DNA are not enough to explain the diversity within individuals of the same species and obviously, because there are different cell types within one individual, not every cell expresses all proteins. Consequently, there have to be additional mechanisms that determine the fate of each cell. One of the most striking of such mechanisms is DNA methylation. It occurs on the cytosine (C) base in the context of CpGs, where a C is followed by a guanine (G) base in the DNA sequence [27, 31, 73]. The conversion of C to 5-methylcytosine (5mC) is carried out by DNA methyltransferase (Dnmt) enzymes [12, 104]. There are two kinds of methylation events, namely maintenance, where existing methylation patterns are reestablished after DNA replication [59] and *de novo*, where new patterns may be introduced [103]. These different kinds of methylation events are mostly (but not exclusively) associated with a certain type of Dnmt [82], for example, Dnmt1 is usually associated with maintenance and Dnmt3a/b with *de novo* methylation.

DNA methylation is known to control and mediate gene expression and therefore varies considerably depending on cell type and genomic locations. Methylation of promoters often correlates with little to no transcription [119] and hence can be used as a predictor of gene expression [66]. This also leads to different methylation levels in different cell types [109], as well as different levels in healthy and cancerous cells [23]. However, the underlying mechanisms that determine the methylation status of specific CpGs and the resulting methy-

lation patterns in different genomic regions are not fully understood. Therefore, mechanistic models to investigate the underlying processes and to estimate the probabilities of the different methylation states of a CpG have been developed [6, 38, 78]. However, these models usually consider only a single CpG or assume an independence of the individual CpGs.

Models that only consider a single CpG or treat the CpGs independently have two major drawbacks: First of all, it is impossible to take possible influences of the neighboring methylation states into account, when only considering one CpG or when considering the CpGs to be independent. The second drawback is, that it is impossible to investigate the possible working mechanisms of the Dnmts, since the sequence in which the CpGs are methylated does not matter in the independent case. We therefore generalize the existing hidden Markov models and take whole methylation patterns into account to tackle both of these problems. To this end we introduce dependence parameters to quantify the influence of the methylation state of neighboring CpGs and propose different transition probabilities (amongst others dependent on the dependence parameters) for all possible methylation states in the direct neighborhood. We also discuss the possible working mechanisms of the Dnmts by manipulating the order of multiplication for the transition matrices for the sub-processes (mainly the order in which CpGs within a sequence are methylated). To generate the transition matrices we rely on a stochastic automata networks description.

To fit the models to the available double-strand methylation data, we suggest different parameter estimation methods. Each of these methods comes with its own advantages and disadvantages and may be appropriate under different circumstances. The focus of the parameter estimations lies on the dependence parameters, in order to investigate the influence of the neighboring methylation states and to infer the possible working mechanisms of the enzymes. We investigate double-strand methylation data for single copy genes and repetitive elements at selected loci, as well as whole genome data. We also test the applicability of our model to single-strand methylation data. Our main results are that Dnmt1 seems to methylate CpGs independent of their neighboring states and appears to work processivly, while there is a dependence on the methylation state of the left (but not the right) neighbor for Dnmt3a/b. Also, hypomethylated CpGs at promoter regions behave more independent compared to hypermethylated CpGs in other regions.

This chapter is organized as follows: In Section 2.2 we give a brief mathematical and biological background and discuss the state of the art related work. Our newly introduced model is thoroughly discussed in Section 2.3. The different techniques that are used for the parameter estimation are briefly described

in Section 2.4. In Section 2.5 we extensively present our findings and conclude this chapter in Section 2.6.

2.2 Background

In this section we lay the necessary mathematical and biological background for the stochastic modelling of spatial methylation patterns.

2.2.1 Mathematical Background

Here, we give a brief overview of the most important mathematical concepts, which are used in this thesis. For a more in-depth introduction to this topic, we refer to common text books like [45, 62, 65].

Stochastic Processes

A *stochastic process* is defined as a collection of random variables (RVs). Thereby a *random variable* on $(\Omega, 2^\Omega, P)$ is a function

$$X : \Omega \to \mathbb{R}, \tag{2.1}$$

where $(\Omega, 2^\Omega, P)$ is a discrete probability space, Ω a sample space, 2^Ω a σ-algebra and P a probability measure. The function

$$\pi : X(\Omega) \to [0, 1] \text{ with } \pi(x) = P(X = x) \tag{2.2}$$

is called the *discrete probability distribution* of X, where $X = x$ is an abridged notation for the set $\{\omega \in \Omega \mid X(\omega) = x\}$, which is a subset of Ω. Subsets obtained from a different relational operator, like $<$ or $\leq$, are defined in a similar way. Subsets of Ω are also called *events*. Given two events $Y = y$ and $X = x$, with $P(X = x) > 0$, the *conditional probability* is defined as

$$P(Y = y \mid X = x) := \frac{P(Y = y \cap X = x)}{P(X = x)}. \tag{2.3}$$

Markov Chains

Markov chains are a special family of RVs X_t, where t is an index. The RVs $X_t : \Omega \to S$ take values in a discrete set S, which is called the *state space*. Note that in this case the state space is countable. Also note that there are generalizations for continuous S, but we do not consider them here. Oftentimes the index

t is associated with time. We therefore call a Markov chain with a discrete index $t \in \mathbb{N}_0$ a *discrete time Markov chain (DTMC)* and a Markov chain with a continuous index $t \in \mathbb{R}_0$ a *continuous time Markov chain (CTMC)*. Since we will only consider DTMCs in the following, we only present the properties of DTMCs here. CTMCs fulfill the same properties with their continuous counterparts.

The special feature of Markov chains is that they fulfill the so-called *Markov property*

$$P(X_{t+1} = y \mid X_t = x_t, X_{t-1} = x_{t-1}, \ldots, X_0 = x_0) = P(X_{t+1} = y \mid X_t = x_t) \quad (2.4)$$

for all $t, y, x_t, \ldots, x_0$. This property implies that only information about the present $X_t = x_t$ is necessary to predict the next step $X_{t+1} = y$. Information about previous states and hence the history of the system are irrelevant to predict the future behavior.

Since $S = \{s_1, s_2, \ldots\}$ is countable, it is possible to enumerate the states, such that we can write $X = i$ instead of $X = s_i$. We can then define the transition probabilities of reaching state j from state i in one step as

$$p_{ij} = P(X_{t+1} = j \mid X_t = i) \quad (2.5)$$

and arrange them into the *transition probability matrix* $P = (p_{ij})_{i,j \in \{1,2,\ldots\}}$. Note that we assume here, that the transition probabilities are time independent, i.e., the same for all t. In that case the process is called *time homogeneous*. The matrix P is a *right stochastic matrix*, which means that it a is square matrix of nonnegative real numbers whose rows add up to 1. Note that the matrix power P^t is right stochastic as well and contains the probabilities to reach another state after t steps.

The *transient distribution* of the Markov chain is defined as the discrete probability distribution of the states at a certain time. Given an *initial distribution*, i.e., a stochastic row vector $\pi(0)$ that contains the probabilities $P(X_0 = i)$ for all states $i \in S$ at time 0, we can use the law of total probability to calculate the probabilities for all states in the next step as

$$P(X_1 = j) = \sum_{i \in S} \underbrace{P(X_1 = j \mid X_0 = i)}_{= p_{ij}} \cdot P(X_0 = i), \ \forall j \in S. \quad (2.6)$$

We can consider Eq. (2.6) as a vector matrix product and write

$$\pi(1) = \pi(0) \cdot P. \quad (2.7)$$

A successive application of this argument leads to the transient distribution at an arbitrary time t

$$\pi(t) = \pi(t-1) \cdot P = \ldots = \pi(0) \cdot P^t. \quad (2.8)$$

Under certain conditions there exists a unique *equilibrium distribution*

$$\pi_e = \pi_e \cdot P \tag{2.9}$$

which is independent from the initial distribution. For a finite DTMC these conditions are *irreducibility* and *aperiodicity*. The equilibrium is characterized by a so-called *global balance* condition, where the inflow of probability mass equals the outflow of probability mass for each state.

Hidden Markov Models

When using DTMCs to model real systems, one often faces the problem that the states are either not directly observable or that the observations are tainted with errors. In this case an appropriate model choice are the so-called *hidden Markov models (HMMs)*. A HMM consists of two stochastic processes: A DTMC X with $X_t : \Omega \to S,\ t \in \mathbb{N}_0$, which models the transitions between the hidden states and a second stochastic process Y with $Y_t : S \to O$, which maps the hidden states from S to the observable states from the (countable) state space of observables O. Note that the probability of having the k-th observation o_k at time t as an output depends only on the hidden state i at time t, i.e.,

$$\Delta_{ik} = P(Y_t = o_k \mid X_t = i) \tag{2.10}$$

and does not depend on the previous states of X. Since we assume this emission probabilities Δ_{ik} to be time independent, we can arrange them in the *emission* or *error matrix* Δ. Note that, as the transition probability matrix P of the DTMC, Δ is a matrix of nonnegative real numbers whose rows add up to 1. However, in contrast to P, the error matrix Δ is not necessarily square since the cardinalities of the hidden state space $|S|$ and the observable state space $|O|$ do not need to coincide.

Typical tasks when working with HMMs involve calculating the probability of a given output sequence, calculating unknown parameters, like the initial distribution of hidden states from a given sequence of observations, or even calculate the most likely hidden sequence. For each of these tasks dedicated algorithms have been developed [107], like the *Viterbi algorithm* [125] to calculate the most likely hidden sequence, given a sequence of observations.

2.2.2 Biological Background

Epigenetics

The term *epigenetics* was first introduced in 1942 by Conrad Waddington [126]. However, the definition was changed and refined over time. The widely accepted definition today defines epigenetics as "the study of changes in gene

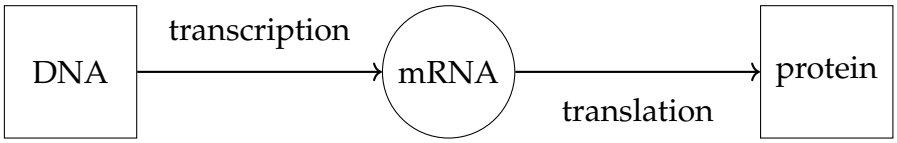


Figure 2.1: Schematic representation of gene expression.

function that are mitotically and/or meiotically heritable and that do not entail a change in DNA sequence" [98].

The DNA contains the blueprints for all proteins (and non-coding, functional RNA) that can be built within an individual. The process of synthesizing a protein from the DNA information is called *gene expression* [1]. A schematic representation of the gene expression is shown in Fig. 2.1. In a first step a certain part of the DNA (gene) is read and transcribed into the so-called messenger RNA (mRNA) with the aid of some special proteins, called transcription factors. Eventually the mRNA is translated into the final protein. Since the DNA sequence itself does not change over time, without additional mechanisms every cell could express all genes. But obviously different cells behave differently and therefore they have to express different genes, while others remain unexpressed. The mechanisms to regulate gene expression [108] are called epigenetic switches.

The three main epigenetic switches (and the intuitive high-level explanations why they shut down gene expression for certain genes) are

- *DNA methylation*: Due to the additional methyl groups on cytosines within promoters, the gene can not be transcribed into mRNA, since the transcription factors can not attach to the DNA [101].

- *histone modification*: Causes loosely or tightly packing of the DNA, thus resulting in expressed or non-expressed genes, respectively [106].

- *mRNA interference*: The mRNA is destroyed due to interaction with other RNA and special enzymes and hence translation can not occur [75].

Since DNA methylation and histone modification repress the transcription of the genes, there is a complete shut down of this gene. For mRNA interference, the gene expression may still occur, however, to a lower extent. Since the destruction of the mRNA depends on the concentration of the other RNA and enzymes, there may still be some translation into proteins, albeit at a lower rate than without interference.

DNA Methylation

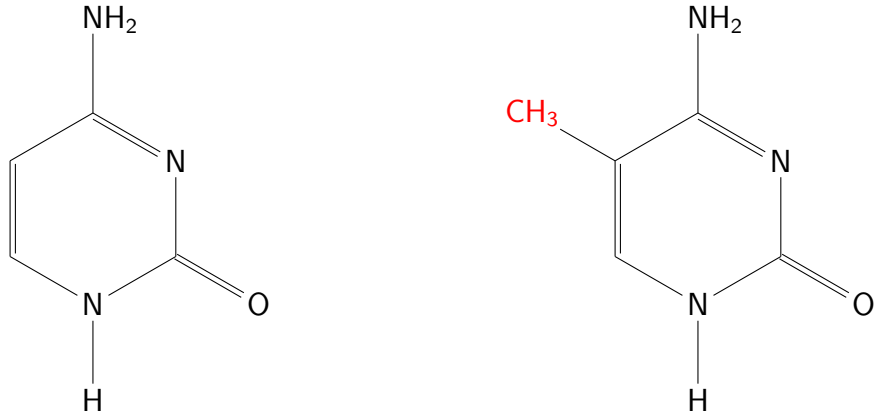


Figure 2.2: Structural formulas of cytosine (left) and 5-methylcytosine (right). They differ in the methyl group CH_3 (marked in red) at the 5 position. Note that we used the skeletal formula, where the carbon atoms and hydrogen atoms bound to them are not explicitly shown.

Since this thesis focuses on DNA methylation, we go a little bit more into detail here. In mammals DNA methylation almost exclusively occurs on *cytosine* (C), more precisely in the context of CpGs [27, 31, 73]. A CpG is a sequence of a C followed by guanine (G) in 5′ to 3′ direction on the DNA. C and G are linked via a phosphate group (p), hence the name CpG. In the double-stranded DNA C is paired with G (and vice versa) on the opposite strand. Therefore, each CpG contains two Cs: one from the CG pair on one strand and one from the respective GC pair on the opposite strand. This symmetry is important for the inheritance of methylation patterns and will be discussed later.

The Cs in CpGs are converted to 5-methylcytosine (5mC) by changing the hydrogen (H) atom at the 5 position to a methyl group CH_3. A structural formula for both C and 5mC can be found in Fig. 2.2. The reaction is controlled by a family of enzymes, the so-called *DNA methyltransferases* (Dnmts) [12, 82, 104]. The mechanisms of how the Dnmts interact with the DNA remain elusive. It is also possible that different kinds of Dnmts behave differently. Two examples of different possible mechanisms are shown in Fig. 2.3: In processive methylation, the Dnmt continuously methylates all CpGs (except for possible errors), while walking in one direction on the DNA strand, here, for example in 5′ to 3′ direction. Distributive methylation is characterized by a constant detachment from and attachment to the DNA, without directed movement. While the Dnmt is attached the C can be methylated and while detached the Dnmt may perform a diffusive motion before attaching at some other sites of the DNA.

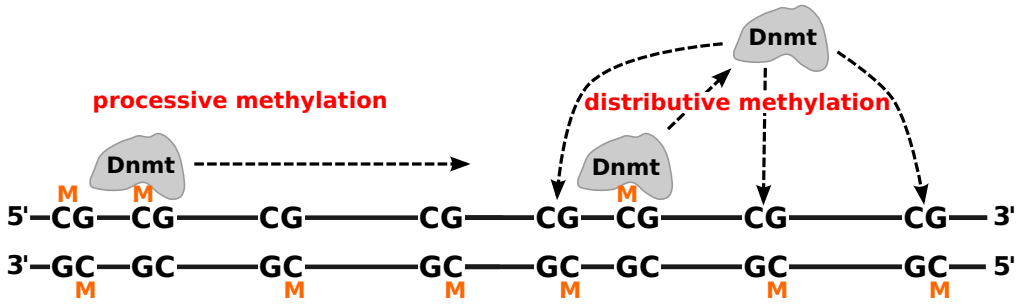


Figure 2.3: Two possible methylation mechanisms. In processive methylation the Dnmts methylate unmethylated Cs in 5′ to 3′ direction without attaching. In distributive methylation the Dnmts attach, perform a methylation event, detach again, move to some other C (not necessarily a direct neighbor) and so forth.

One distinguishes different kinds of methylation events, which are (for the most part) associated with different members of the Dnmt family. *Maintenance* methylation is responsible to reestablish methylation patterns after cell divisions and is mainly associated with Dnmt1 [59]. After replication one strand (and its methylation) is kept as it is (parental strand), while the opposite strand is newly synthesized (daughter strand). However, on the new strand all Cs are initially unmethylated. Therefore, all CpGs are either un- or hemimethylated, i.e., only the C on the parental strand within the CpG is methylated. Note that hemimethylation can occur for both strands, independent of their role of parental or daughter strand. However, immediately after replication only the parental strand may contain a methylated C.

We now call the methylation events after replication, which make hemimethylated CpGs fully methylated, maintenance methylation. Essentially, in maintenance methylation the information about the methylation state of the parental strand is used to reestablish existing methylation patterns after cell division, i.e., if the C on the parental strand is methylated, then the C on the daughter strand will (very likely) also be methylated and if the C on the parental strand is unmethylated, no methylation event will take place. Therefore the maintenance probability is usually very high [6]. Here also the aforementioned symmetry in CpGs comes into play: If methylation would occur on single Cs (outside of the CpG context), after cell division the information about the methylation state would only be conserved on the (parental) strand containing the C. For the other (parental) strand containing the G, the newly synthesized daughter strand will contain an unmethylated C, however there is no way of inferring whether the C should be methylated or not to preserve the initial pattern.

In *de novo* methylation, on the other hand, which is carried out by Dnmt3a and Dnmt3b (which we summarize to Dnmt3a/b and treat them as one), methyl groups may be added to any arbitrary unmethylated C, independent of the methylation state of the opposite strand [103]. Hence, *de novo* methylation may be responsible for two tasks: First to add new methylation patterns, if a fully unmethylated CpG is methylated on one (or both) strand(s) and second to fix a failed maintenance if methylation is added to a hemimethylated CpG when the maintenance methylation was not performed before. Usually the *de novo* probability is quite small compared to the maintenance probability [6], since once the cell differentiation is complete, the methylation patterns are quite stable. Note that there are sometimes different definitions for the methylation events, where *de novo* is exclusively the transition from un- to hemimethylated and not the maintenance-like transitions.

Also note that the maintenance and *de novo* efficiencies are not necessarily constant over time [38]. Since the focus of this thesis lies more on the spatial modelling and neighborhood dependencies for methylation events, we keep the efficiencies constant in order to not make the model too complex. Time dependent transition probabilities (efficiencies) require more parameters and transition matrices and tend to be prone to overfitting, if not treated carefully.

Recent studies suggest, that although Dnmt1 is mainly responsible for maintenance, it may also perform *de novo* methylation to a certain degree. Also, Dnmt3a/b, mainly responsible for *de novo*, may perform some maintenance [82]. Therefore, when modelling the enzymes and methylation events, each enzyme should get its own parameter set. With the aid of genetic engineering, it is possible to knock out (KO) one of the enzymes such that only the other is active and hence allows to investigate the properties of the different enzymes separately. In case of Dnmt1KO only Dnmt3a/b is active and in case of Dnmt3a/b DKO (double KO, since two enzymes are knocked out at once) only Dnmt1 is active. The case where no enzymes are knocked out is referred to as *wild-type*. The data sets are obtained from so-called hairpin bisulfite sequencing (BS-seq) and will be discussed in more detail in Section 2.3.7.

It is important to note that 5mC can be further modified by oxidation to 5-hydroxymethyl- (5hmC), 5-formyl- (5fC) and 5-carboxylcytosine (5caC) by Tet enzymes [69]. These modifications are involved in the removal of 5mC from the DNA by passive or active demethylation. Passive demethylation is connected to DNA replication, when a new strand is synthesized with no methylation (or other modifications) at all. Due to a low maintenance efficiency, or due to the fact that the modifications of 5mC are not recognized on the opposite strand for maintenance events, the methylation level subsequently decreases (passively) over the course of some cell divisions. In active demethylation, on the other

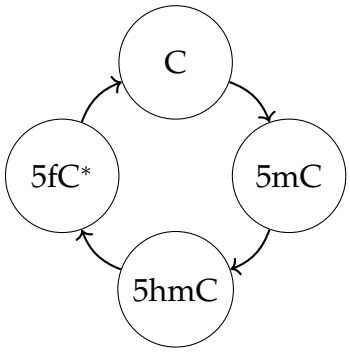


Figure 2.4: Schematic representation of *de novo* methylation and the active demethylation loop. Note that we summarized 5fC and 5caC into 5fC*.

hand, 5mC is eventually modified to 5fC* (5fC or 5caC), which is actively removed from the DNA and replaced by an unmethylated C. A schematic representation of this methylation cycle is depicted in Fig. 2.4. Note that this cycle can occur multiple times between two cell divisions[1] and therefore requires hybrid modelling [77]. However, the data used for this thesis does not capture modifications other than 5mC. Therefore, we do not consider the other modifications here and stick to modelling in discrete time.

2.2.3 Mechanistic Modelling vs. Machine Learning

> *"All models are wrong, but some are useful."*
>
> (George E. P. Box)

The goal of *mechanistic models* is to gain understanding of the underlying processes in real systems. To that end, the model approximates the real system by focusing on the relevant/interesting aspects, usually using a mathematical description, and omitting the non-relevant parts. It is important to note, that no matter how complex the model becomes, certain aspects of the real system will always be missed or cannot be described by the model. Hence, "all models are wrong". Nevertheless, these models have successfully been applied to describe certain aspects of the real system and to get a deeper understanding, and "are [therefore] useful". In physics, for example, there are different models that describe different aspects of the same system (the universe): Processes on a very small scale are governed by quantum mechanics, while on a large scale gravity is the appropriate model. Both modelling approaches work very well

[1] Since maintenance happens only once after cell division, the cycle is for the vast majority of runs driven by *de novo* methylation.

in their respective case, however, there is no model (yet), that is able to describe processes for all scales, small and large.

Mechanistic models usually contain parameters or assumptions that directly allow an interpretation to explain the behavior of the system. To be a bit more precise and establish ties to this thesis, in Section 2.3 we will introduce a mechanistic model uses both aforementioned possibilities: The model contains so-called *dependency parameters* that directly allow to interpret the strength of the influence of the neighboring methylation states on the transition rates. At some point we also have to choose the multiplication order of the transition matrices, which allows us to investigate different assumptions on the working mechanisms of how the Dnmts interact with the DNA (processive vs. distributive methylation, see also Fig. 2.3).

In *machine learning*, on the other hand, it is usually not possible to use the obtained results to explain the system's behavior. Only the simplest techniques, like subset selection, decision trees or linear regression, allow some interpretability of the results, however, these methods are quite inflexible and therefore often not suitable to produce reliable predictions for more complex scenarios.

The goal of machine learning is to build a statistical model from data, based on generalizable rules and patterns. This model can then be used to make predictions on new unseen data. However, the underlying algorithms function in a black box manner, i.e., they usually yield good prediction results, but how these results are obtained is usually not explainable. This is a sharp contrast to mechanistic models. Although this thesis focuses mainly on mechanistic modelling, whenever appropriate we borrow some machine learning methods, as for example in the clustering in Section 2.5.5 that is based on a variant of k-means. In general, it could be a promising approach to combine the best of this two worlds: the explainability from mechanistic modelling with the predictive power from machine learning.

2.2.4 Related Work

With the rapid evolution of high-throughput technologies for epigenetic analysis, data on a genome-wide scale is available [15, 16, 39, 80, 91] and computational methods are contributing significantly to the progress of epigenetic research. For instance, deep learning can be used to impute the methylation state at individual DNA positions if information about the state of neighboring positions is available [5]. Another approach to impute on unassayed CpG sites is based on Bayesian clustering [67]. As an orthogonal approach to learning-based methods, which focus on accurate predictions, mechanistic models have been

developed to describe the mechanisms underlying epigenetic changes and test different hypotheses [33, 36, 105, 117].

Arand et al. proposed a hidden Markov model (HMM) for the evolution of DNA methylation patterns during early development and applies it to hairpin bisulfite sequencing data from mouse embryonic stem cells [6]. It gives a mechanistic description of the activity of the DNA methyltransferases Dnmt1, Dnmt3a, and Dnmt3b over time, as well as the loss of methyl groups through cell division. Since in mammals, DNA methylation primarily occurs on the cytosine nucleotide of a CpG site, the model considers the methylation state of individual CpGs over time. Trained on KO data, the model is able to predict unseen methylation patterns in wild-type. A similar model has been used to gain insights into the detailed molecular mechanisms underlying passive and active demethylation [77]. Moreover, for genome-wide data, parameter values that describe the efficiency of epigenetic modifications in such models can be clustered and correlated with data from enrichment analysis [76].

Several mechanistic models have been proposed that consider methylation patterns of some successive CpGs and their spatial relationships [14, 35, 48, 79, 84, 96]. Here, we consider a spatial extension of the model considered in [6]. Its main strength compared to other models is that for each locus, it considers methylation efficiencies and dependency parameters. Moreover, it describes the methylation state of both DNA strands and is thus appropriate for data from hairpin bisulfite sequencing [40].

A major challenge is that the complexity of models considering methylation patterns of several CpGs is much higher than the complexity of models that consider a CpG in isolation. In the former case, all possible combinations of states of the individual CpGs have to be considered during the analysis. Standard numerical approaches for parameter estimation based on maximizing the likelihood of the data [6] fail for such models, since the number of possible states is too large. Likelihood-free approaches based on stochastic sampling, such as Approximate Bayesian Computation have been applied in this context [14]. They allow to estimate the posterior distribution based on a comparison of measured and simulated data sets but often suffer from slow convergence to the true posterior distribution.

In [14] location- and neighbor-dependent models are proposed for single-stranded DNA methylation data in blood and tumor cells. The (de-)methylation rates depend on the position of the CpG relative to the 3′ or 5′ end and/or on the methylation state of the left neighbor only. The dependence is realized by the introduction of an additional parameter. In our proposed models we use double-stranded DNA and can, therefore, include hemimethylated sites and even distinguish on which strand the site is methylated. Furthermore, we allow

dependencies on both neighbors by introducing two different dependence parameters. In contrast [35] copes with the neighborhood dependence indirectly by allowing different parameter values for different sites. In order to reduce the dimensionality of the parameter vector, a hierarchical model based on beta distributions is proposed. Another difference to our model is the distinction between *de novo* rates for parent and daughter strand. However, this can easily be included in future work.

A density-dependent Markov model was proposed [79]. In this model, the probabilities of (de-)methylation events may depend on the methylation density in the CpG neighborhood. In addition, a neighboring sites model has been developed, in which the probabilities for a given site are directly influenced by the states of neighboring sites to the left and right [79]. When these models were tested on double-stranded methylation patterns from two distinct tandem repeat regions in a collection of ovarian carcinomas, the density-dependent and neighboring sites models were superior to independent models in generating statistically similar samples. Although this model also includes the dependence on the methylation state on the left and right neighbor for double-stranded DNA the approach is different. The transition probabilities of the neighbor-independent model are transformed into a transition probability of a neighbor-dependent model by introducing only one additional parameter. The state of the left and right neighbor are taken into account by exponentiating this parameter by some norm. In addition, this approach does not allow the intuitive interpretation of the dependence parameter. Recently the model from [79] was extended to include the influence of different distances between the CpGs [96]. However this model is still restricted to single-stranded methylation data.

In [48] it has been shown that the collaboration between CpG sites is required to obtain stable fractions of methylation states over time in CpG islands (CGIs). In this model another nearby CpG serves as a mediator such that its state influences the possible reactions. In a more recent version of this model the distance to the mediator CpG is taken into account [84]. However, both models feature active demethylation, but have no explicit dependence parameter and do not distinguish between the two different hemimethylated states.

Another approach based on the Ising model from statistical physics is introduced in [64]. The correlation between methylation states are accounted for by using a joint probability model. Recently, [47] introduced a model that contains changes in the methylation states for both single sites and whole CGIs. However, the efficiency of this model is based on the assumption of the conditional independence of CpG sites and therefore does not take into account the (possible) correlations between these CpG sites. Another recent approach combines statistical inference with mathematical modelling [21]. At first they infer the

parameters for each CpG site independently and investigate correlations with the genomic distance as well as the influence of the local CpG density. In a next step they use the obtained parameters for a stochastic simulation and compare the results to real data.

2.3 Model

2.3.1 Notation

Consider a sequence of L neighboring CpG dyads[2], which is represented as a lattice of length L and width two (for the two strands). Each cytosine in the lattice can either be methylated or not, leading to four possible states at each position l:

- *State 0*: Both cytosines are not methylated.
- *State 1*: The cytosine on the upper strand is methylated, the lower one not.
- *State 2*: The cytosine on the lower strand is methylated, the upper one not.
- *State 3*: Both cytosines are methylated.

A sequence of four CpGs, each of which is in one of the four possible states, is shown in Fig. 2.5. For a system of length L there are in total 4^L possibilities to combine the states of individual CpGs. These combinations are called *patterns* in the following. A pattern is denoted by a concatenation of states $s_1 s_2 \dots s_L$, e.g. 321 or 0123.

In order to represent the pattern distribution as a vector it is necessary to uniquely assign a reference number Z to each pattern. A pattern can be perceived as a number in the tetral system, such that converting to the decimal

[2]The exact nucleotide distance between two neighboring dyads is not considered here explicitly, but we assume that this distance is small. For the BS-seq data that we consider, the average distance between two CpGs is 14 bps (base pairs) and the maximal distance is 46 bps.

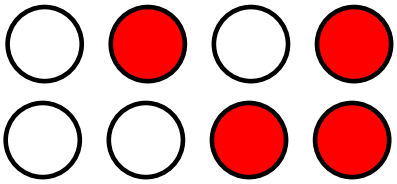


Figure 2.5: A lattice of length $L = 4$ containing all possible states 0, 1, 2 and 3, forming the pattern $s_1 s_2 s_3 s_4 = 0123$.

system leads to a unique reference number. After the conversion an additional 1 is added in order to start the referencing at 1 instead of 0, i.e.

$$Z = 1 + \sum_{l=1}^{L} s_l \cdot 4^{l-1}. \tag{2.11}$$

For example, for $L = 3$:

$$\begin{aligned} \text{pattern: } 000 &\longrightarrow Z = 1 \\ \text{pattern: } 123 &\longrightarrow Z = 28 \\ \text{pattern: } 333 &\longrightarrow Z = 64 \end{aligned}$$

This reference number Z then also corresponds to the position of the pattern in the respective distribution vector. Note that this definition of Z is only used to uniquely enumerate the patterns and allows the convenient transformation from pattern to reference number and back. The emerging ordering with respect to the reference number does not necessarily reflect the (dis)similarity of different patterns. For example, the patterns 002 and 100, which are essentially the same pattern (only considering CpGs and ignoring the rest of the DNA), i.e, containing exactly one methylated C at the respective 5' end, have reference numbers 3 and 17 for $L = 3$. Hence, they are quite far apart in the distribution vector. Thus, the reference number Z falls in the category of qualitative, nominal data and, therefore, classical moments of the distribution with respect to Z, like the average pattern, have no intuitive meaning.

We describe the state of a sequence of L CpGs by a discrete-time Markov chain with pattern distribution $\pi(t)$, i.e., the probability of each of the 4^L patterns after t cell divisions. For the initial distribution $\pi(0)$, we use the distribution measured in the wild-type when the cells are in equilibrium. Note, that other initial conditions gave very similar results, i.e., the choice of the initial distribution does not significantly affect the results. The reason is that also the KO data is measured after a relatively high number of cell divisions where the cells are almost in equilibrium. Transitions between patterns are triggered by different processes: First due to *cell division* the methylation on one strand is kept as it is (e.g. the upper strand), whereas the newly synthesized strand (the new lower strand) does not contain any methyl group. Afterwards, methylation is added due to different mechanisms. On the newly synthesized strand, a site can be methylated if the cytosine at the opposite strand is already methylated (*maintenance*). It is widely accepted that maintenance in form of Dnmt1 is linked to the replication machinery and thus occurs during/directly after the synthesis of the new strand. Furthermore, CpGs on both strands can be methylated independent of the methylation state of the opposite site (*de novo*). The

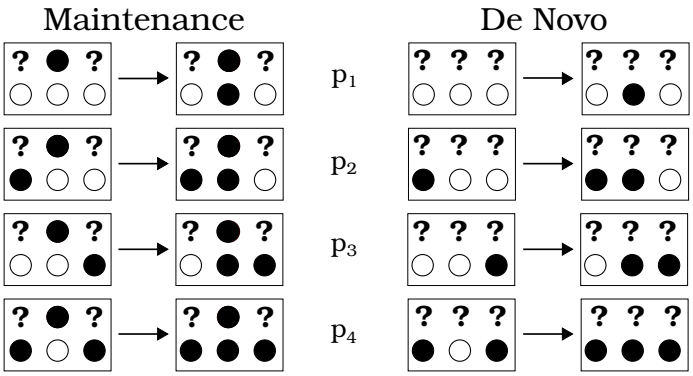


Figure 2.6: Possible maintenance and *de novo* transitions depicted for the lower strand, where ○ denotes an unmethylated, ● a methylated site and **?** a site where the methylation state does not matter. Note that the same transitions can occur on the upper strand.

transition matrix P is defined by the composition of matrices for cell division, maintenance and *de novo* methylation of each site.

2.3.2 Cell Division

Depending on which daughter cell is considered after cell replication, the upper ($s = 1$) or lower ($s = 2$) strand is the parental one after cell division. Then, the new pattern can be obtained by applying the following state replacements:

$$s = 1: \begin{cases} 0 & \longrightarrow & 0 \\ 1 & \longrightarrow & 1 \\ 2 & \longrightarrow & 0 \\ 3 & \longrightarrow & 1 \end{cases} \qquad s = 2: \begin{cases} 0 & \longrightarrow & 0 \\ 1 & \longrightarrow & 0 \\ 2 & \longrightarrow & 2 \\ 3 & \longrightarrow & 2 \end{cases} \tag{2.12}$$

Given some initial pattern with reference number i, applying the transformation (2.12) to each of the L positions leads to a new pattern with reference number j (notation: $i \overset{(2.12)}{\rightsquigarrow} j$). The corresponding transition matrix $CD_s \in \{0,1\}^{4^L \times 4^L}$ has the form

$$CD_s(i,j) = \begin{cases} 1, & \text{if } i \overset{(2.12)}{\rightsquigarrow} j, \\ 0, & \text{else.} \end{cases} \tag{2.13}$$

2.3.3 Maintenance and *De Novo* Methylation

For maintenance and *de novo* methylation, the single site transition matrices are built according to the following rules:

Consider at first the (non-boundary) site $l = 2, \ldots, L-1$ and its left and right neighbor $l-1$ and $l+1$ respectively. The remaining sites do not change and do not affect the transition. The probabilities of the different types of transitions in Fig. 2.6 have the form

$$p_1 = 0.5 \cdot (\psi_L + \psi_R)x, \tag{2.14}$$
$$p_2 = 0.5 \cdot (\psi_L + \psi_R)x + 0.5 \cdot (1 - \psi_L), \tag{2.15}$$
$$p_3 = 0.5 \cdot (\psi_L + \psi_R)x + 0.5 \cdot (1 - \psi_R), \tag{2.16}$$
$$p_4 = 1 - 0.5 \cdot (\psi_L + \psi_R)(1 - x), \tag{2.17}$$

where we set the probability x to $x = \mu$ in case of maintenance and to $x = \tau$ in case of *de novo* methylation. $\psi_L,\ \psi_R \in [0, 1]$ are the dependence parameters for the left and right neighbor. The probabilities correspond to the four possible cases for the neighbor states: p_1 is used when both neighbors are unmethylated, p_2 if only the left, p_3 if only the right, and p_4 if both neighbors are methylated.

A dependence value of $\psi_i = 1$ corresponds to a total independence on the neighbor whereas $\psi_i = 0$ leads to a total dependence. Hence, μ and τ can be interpreted as the probability of maintenance and *de novo* methylation of a single cytosine between two cell divisions assuming independence from neighboring CpGs. A visualization of the transition probabilities (2.14) - (2.17) can be found in Fig. 2.7. Moreover, all CpGs that are part of the considered window of the DNA have the same value for the parameters μ, τ, ψ_L, and ψ_R, since in earlier experiments only very small differences have been found between the methylation efficiencies of nearby CpGs [6].

In order to understand the form of the transition probabilities consider at first a case with only one neighbor. The probabilities then have the form ψx if the neighbor is unmethylated and $1 - \psi(1 - x)$ if the neighbor is methylated. Note that both forms evaluate to x for $\psi = 1$, meaning that a site is methylated with probability x, independent of its neighbor. For $\psi = 0$ the probabilities become 0 and 1, meaning that if there is no methylated neighbor the site cannot be methylated or will be methylated for sure if there is a methylated neighbor respectively.

The probabilities for two neighbors are obtained by a linear combination of the one neighbor cases, with ψ_L for the left and ψ_R for the right neighbor, and an additional weight of 0.5 to normalize the probability. The same considerations also apply to the boundary sites however there is no way of knowing the methylation states outside the boundaries (denoted by ?). Therefore instead of a concrete methylation state ($\circ$ for unmethylated, $\bullet$ for methylated site) the average methylation density ρ is used to compute the transition probabilities at

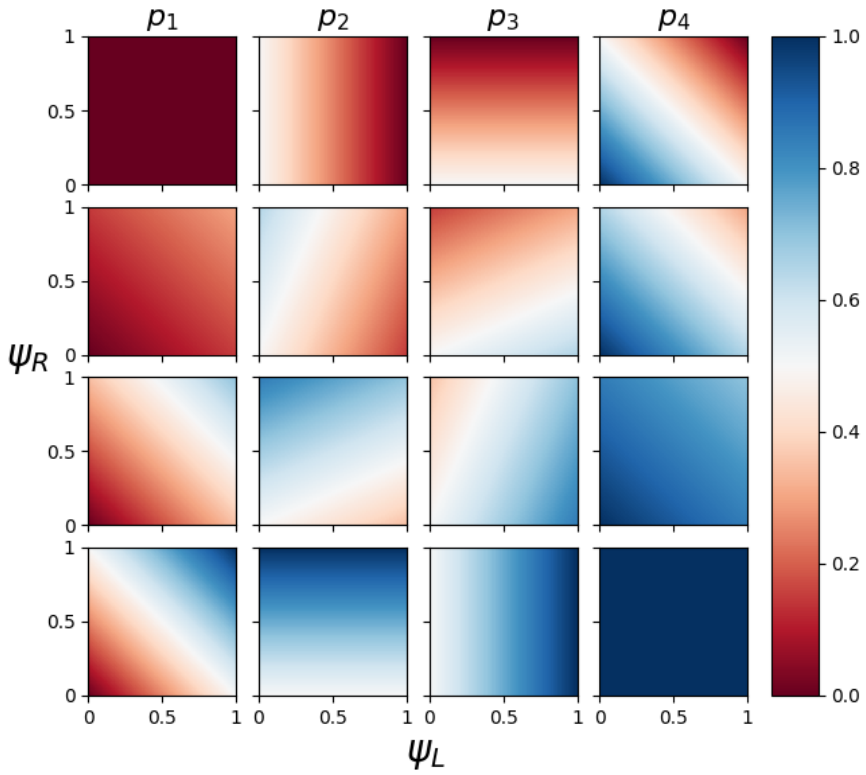


Figure 2.7: Transition probabilities p_1 - p_4 (left to right) for $x = 0, 0.3, 0.7, 1$ (top row to bottom row).

the boundaries (depicted here for *de novo*):

$$? \circ \circ \rightarrow ? \bullet \circ \qquad \tilde{p}_1 = (1-\rho)\cdot p_1 + \rho \cdot p_2, \tag{2.18}$$

$$? \circ \bullet \rightarrow ? \bullet \bullet \qquad \tilde{p}_2 = (1-\rho)\cdot p_3 + \rho \cdot p_4, \tag{2.19}$$

$$\circ \circ ? \rightarrow \circ \bullet ? \qquad \tilde{p}_3 = (1-\rho)\cdot p_1 + \rho \cdot p_3, \tag{2.20}$$

$$\bullet \circ ? \rightarrow \bullet \bullet ? \qquad \tilde{p}_4 = (1-\rho)\cdot p_2 + \rho \cdot p_4. \tag{2.21}$$

Note that the same considerations hold for maintenance at the boundaries if the opposite site of the boundary site is already methylated.

For each position l, there are four transition matrices: two for maintenance and two for *de novo*, namely one for the upper and one for the lower strand in each process. In order to construct these matrices consider the three positions $l-1$, l and $l+1$, where the transition happens at position l. Only the transitions depicted in Fig. 2.6 can occur. Furthermore the transitions are unique, i.e. for a given reference number i the new reference number j is uniquely determined. For patterns not depicted in Fig. 2.6 no transition can occur, i.e. the reference number does not change.

The matrix describing a maintenance event at position l and strand s has the form

$$M_s^{(l)}(i,j) = \begin{cases} 1, & \text{if } i = j \text{ and } \nexists j' : \ i \rightsquigarrow j', \\ 1-p, & \text{if } i = j \text{ and } \exists j' : \ i \rightsquigarrow j', \\ p, & \text{if } i \neq j \text{ and } i \rightsquigarrow j, \\ 0, & \text{else,} \end{cases} \tag{2.22}$$

where the probability p is given by one of the Eqs. (2.14)-(2.21) that describes the corresponding case and $x = \mu$. Note that $M_s^{(l)}$ depends on s and l since it describes a single transition from pattern i to pattern j, which occurs on a particular strand and at a particular location with probability p. We define matrices $T_s^{(l)}$ for *de novo* methylation according to the same rules except that $x = \tau$ and the possible transitions are as in Fig. 2.6, right. All matrices are of size $4^L \times 4^L$. The advantage of defining the matrices position- and process-wise is that different models can be realized by changing the order of multiplication of these matrices.

It is important to note that we have a $4^L \times 4^L$ transition matrix for each strand and position here, i.e., each matrix describes the transition of exactly one C for a given process. Hence, with different multiplication orders, different assumptions about working mechanisms of the Dnmts can be realized as demonstrated in the following section. In Section 2.3.5 we present an alternative approach to generate the transition matrix, where all Cs on one strand are updated simultaneously with only one matrix per process.

2.3.4 Combination of Transition Matrices

For all subsequent models it is assumed that first of all cell division happens and maintenance methylation only occurs on the newly synthesized strand given by s, whereas *de novo* methylation happens on both strands. Given the mechanisms in Fig. 2.3, the two different kinds of methylation events, and the two types of enzymes, there are several possibilities to combine the transition matrices. We consider the following four models, which we found most reasonable based on the current state of research in DNA methylation:

1. first processive maintenance and then processive *de novo* methylation

$$P_s = \prod_{l_1=1}^{L} M_s^{(l_1)} \prod_{l_2=1}^{L} T_1^{(l_2)} \prod_{l_3=1}^{L} T_2^{(l_3)}, \tag{2.23}$$

2. first processive maintenance and then *de novo* in arbitrary order

$$P_s = \frac{1}{(L!)^2} \prod_{l_1=1}^{L} M_s^{(l_1)} \left(\sum_{\sigma_1 \in S_L} \prod_{l_2=1}^{L} T_1^{(\sigma_1(l_2))} \right) \left(\sum_{\sigma_2 \in S_L} \prod_{l_3=1}^{L} T_2^{(\sigma_2(l_3))} \right), \tag{2.24}$$

3. maintenance and *de novo* at one position, processive

$$P_s = \prod_{l=1}^{L} M_s^{(l)} T_1^{(l)} T_2^{(l)}, \tag{2.25}$$

4. maintenance and *de novo* at one position, arbitrary order

$$P_s = \frac{1}{L!} \sum_{\sigma \in S_L} \prod_{l=1}^{L} M_s^{(\sigma(l))} T_1^{(\sigma(l))} T_2^{(\sigma(l))}, \tag{2.26}$$

where S_L is the set of all possible permutations for the numbers $1, \ldots, L$. We will call the resulting models from Eqs. (2.23)-(2.26) *model 1-4* in the following, according to their enumeration above.

Note that the *de novo* events on both strands are independent, i.e. the *de novo* events on the upper strand do not influence the *de novo* events on the lower strand and vice versa, such that $[T_1^{(l)}, T_2^{(l')}] = 0$ independent of ψ_i[3]. Obviously it is important whether maintenance or *de novo* happens first, since the transition probabilities and the transitions themselves depend on the actual pattern. Furthermore in the case $\psi_i < 1$ (dependence on right and/or left neighbor) the order of the transitions on a strand matters, i.e. $[M_s^{(l)}, M_s^{(l')}] \neq 0$ and $[T_s^{(l)}, T_s^{(l')}] \neq 0$ for $l \neq l'$. Note that this definition of models in principle allows to consider an arbitrary number of CpGs. However, at least three CpGs are needed to properly include the influence of the left and right neighbor in the transitions. It is also important to note that independent of the number of considered CpGs the window size of the influential CpGs for the transition rates is always kept at size three. However, treating more than three CpGs at once has two major drawbacks: First of all the number of possible patterns grows rapidly (recall 4^L possible patterns for L CpGs) and hence the transition matrices become very large as well ($4^L \times 4^L$). This may lead to memory issues while calculating the distributions, which can however be circumvented by sampling approaches, i.e. stochastic simulation of the underlying Markov chain. Another problem with the large number of possible patterns is that more data is required in order to

[3] $[A, B] = AB - BA$ is the commutator of the matrices A and B.

ensure a good coverage, i.e. the number of measurements should be larger than the number of patterns.

The second main problem is that using the same dependence parameters for all pairs of adjacent CpGs is a rather strong assumption. Note that this assumption becomes more problematic for larger windows, due to e.g. different distances between the CpGs. One solution would be to introduce extra dependence parameters for each pair, however this may lead to difficulties in the parameter identification.

The total transition matrix is then given by a combination of the cell division and maintenance/*de novo* matrices. Recall that we consider two different types of Dnmts, i.e., Dnmt1 and Dnmt3a/b. If only one type of Dnmt is active (KO data) the matrix has the form

$$P = 0.5 \cdot (CD_1 \cdot P_1 + CD_2 \cdot P_2) \tag{2.27}$$

and if all Dnmts are active (WT data)

$$P = 0.5 \cdot (CD_1 \cdot P_1 \cdot \tilde{P}_1 + CD_2 \cdot P_2 \cdot \tilde{P}_2), \tag{2.28}$$

where P_s and $\tilde{P}_s$ have one of the forms (2.23)-(2.26). This leads to four different models for one active enzyme or 16 models for all active enzymes respectively. In the second case P_s represents the transitions caused by Dnmt1 and $\tilde{P}_s$ the transitions caused by Dnmt3a/b. Note that if $\psi_L = \psi_R = 1$ all models are the same within each case since they reduce to the neighborhood independent model from [6]. Furthermore, the cell division, maintenance, and *de novo* transition matrices for a single CpG at a given position are sparse. However, upon combining them to the full transition matrices in Eqs. (2.27) or (2.28), the final matrices become dense and therefore have higher memory requirements. Note that with increasing L the density of P decreases (see also Fig. 2.8), however, the number of non-zero entries and thus the memory requirements still increase.

2.3.5 Stochastic Automata Network Description

Instead of manually generating the transition matrices for L CpGs, we present a formal approach here, which enables us to generate transition matrices for an arbitrary number of CpGs from the single CpG matrices.

The transition probability matrices for cell division, maintenance and *de novo* methylation for a single CpG are listed in Tab. 2.1. In the left column the matrices concerning the upper strand are listed and in the right column the matrices for the lower strand.

It is biologically plausible to assume that cell division happens first, afterwards maintenance on the daughter strand and in the end *de novo* on both

Table 2.1: Transition matrices for a single CpG. Note that the transition probabilities f may be functions of the reaction parameters, the CpG position and/or the states of the adjacent CpGs. The matrices in the left column represent the transitions on the upper and the matrices in the right column the transitions on the lower strand.

Cell Division

$$CD_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad CD_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Maintenance

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1-f & f \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad M_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1-f & 0 & f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

De Novo

$$T_1 = \begin{pmatrix} 1-f & f & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1-f & f \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad T_2 = \begin{pmatrix} 1-f & 0 & f & 0 \\ 0 & 1-f & 0 & f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

strands takes place. Note that the order of the two possible *de novo* events does not matter, i.e. $T_1 \cdot T_2 = T_2 \cdot T_1$. Since the strand that is kept after cell division is chosen randomly with equal probability, the total transition probability matrix P for one acting enzyme (see also Eq. (2.27)) is given by

$$P = 0.5 \cdot (CD_1 \cdot M_1 + CD_2 \cdot M_2) \cdot T_1 \cdot T_2. \tag{2.29}$$

Given a sequence of L CpGs, each CpG can be described by the aforementioned DTMC, which gives us one automaton of the stochastic automata network (SAN). The structure of each automaton is independent of the automata describing neighboring CpGs, however, the transition probabilities may depend on their local states (functional transitions). A suitable method to combine these automata in order to capture the dynamics of whole sequences of CpGs is to consider them as an automata network. Within the SAN framework,

the transition matrices of the individual automata are combined via the Kronecker product. Since in our case the transition matrix for one automaton is a product of the transition matrices for the different processes, we exploit some properties of the Kronecker product to generate the global transition matrix of the network. From

$$A \otimes (B \otimes C) = (A \otimes B) \otimes C \tag{2.30}$$
$$(AC) \otimes (BD) = (A \otimes B) \cdot (C \otimes D) \tag{2.31}$$

the following properties can be derived [25]

$$\left(\prod_{n=1}^{N} A_n\right) \otimes \left(\prod_{n=1}^{N} B_n\right) = \prod_{n=1}^{N} (A_n \otimes B_n)\,, \tag{2.32}$$
$$\bigotimes_{m=1}^{M} \left(\prod_{n=1}^{N} A_n^{(m)}\right) = \prod_{n=1}^{N} \left(\bigotimes_{m=1}^{M} A_n^{(m)}\right). \tag{2.33}$$

Note that in Eqs. (2.31)-(2.33) the corresponding matrices have to be compatible under the standard matrix product.
As a consequence of Eq. (2.33) it is possible to obtain the total transition matrix P in two ways: First compute a transition matrix for a single CpG (Eq. (2.29)) and extend the result to several CpGs with the Kronecker product or calculate the transition matrices for the different processes for several CpGs first via the Kronecker product and combine them afterwards. Since the transition probabilities may depend on the neighbor states, i.e. the states of the adjacent automata, it is easier to choose the second possibility and construct the individual transition matrices for the different processes first. Another advantage is that the matrices for the different processes are sparse, while the total transition matrix is quite dense for a single CpG (see Tab. 2.1 and Fig. 2.8), such that we apply the Kronecker product to sparse matrices and multiply the (also sparse) results afterwards.

If we assume that all CpGs are methylated independent of their neighborhood, then no functional transitions are needed and the transition probabilities are constants. The construction of the global transition matrix is then straightforward by simply applying the Kronecker product. To model dependence, first observe that since only the transition probabilities, but not the transitions themselves, depend on the neighboring states, the structure of the global transition matrix is the same as in the independent case. By using functions instead of constant probabilities, we are able to capture the effect of the neighbors on the transition rates. Another advantage of the functions is that we can incorporate different model assumptions (like processivity) by using different functions

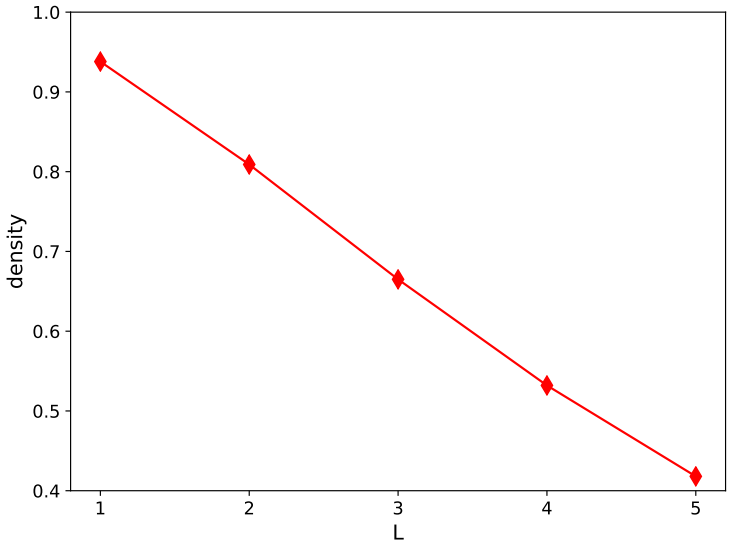


Figure 2.8: Density of entries for the transition matrix P for different numbers of CpGs L.

without altering the structure of the transition matrices.
To shape the function $f := f(\vec{r}, l, s_{l-1}, s_{l+1}) \in [0, 1]$ in the matrices in Tab. 2.1 to our needs, we use the following inputs:

- $\vec{r}$ is a vector with the reaction parameters,
- $l \in \{1, \ldots, L\}$ is the position of the CpG such that boundary ($l = 1, L$) and non-boundary ($l = 2, \ldots, L-1$) CpGs can be distinguished,
- s_{l-1} is the state of the left neighboring CpG and
- s_{l+1} is the state of the right neighboring CpG.

Depending on the methylation event (maintenance or *de novo*) different parameter vectors $\vec{r}$ can be chosen. Since in general all CpGs may undergo a reaction, the states of the neighboring CpGs that are used (before or after reaction) as an input for the function depends on the underlying assumptions. This will be demonstrated in the following.

We first note that the indices of the matrices in Tab. 2.1 correspond to the states before and after transition, i.e. the entry $a_{i,j}$ corresponds to the probability of going from state i to state j. Furthermore, there is a unique relation between the indices of the initial matrices and the indices of the result of their Kronecker

product

$$a_{r,s} \cdot b_{v,w} = (A \otimes B)_{p(r-1)+v,q(s-1)+w}, \tag{2.34}$$

$$(A \otimes B)_{i,j} = a_{\lfloor (i-1)/p \rfloor + 1, \lfloor (j-1)/q \rfloor + 1} \cdot b_{i - \lfloor (i-1)/p \rfloor p, j - \lfloor (j-1)/q \rfloor q}, \tag{2.35}$$

where A is a $p \times q$- and B an arbitrary matrix. These formulas can easily be generalized such that for a Kronecker product of L matrices we know exactly the indices for each of the matrices and therefore the states before and after transition for each CpG. We then use this knowledge to choose the correct transition probability depending on our assumptions. Note that the resulting indices correspond exactly to our previous definition in Eq. (2.11), i.e. they are the corresponding decimal number (shifted by one) when considering a state as a number in the tetral system. For the transition probabilities, we use Eqs. (2.14) - (2.21), depending on which is the appropriate choice for the given assumption and position in the transition matrix.

For a moderate number of CpGs (≈ 5) it is possible to explicitly construct the whole transition matrix with a simple algorithm. We first note that we have to apply the Kronecker product for the matrices in Tab. 2.1 L times with themselves for a sequence of L CpGs. We then apply the following scheme:

1. Identify the indices of the non-zero entries of the matrix.

2. Calculate the indices of the resulting matrix after applying the Kronecker product with Eq. (2.34) for the indices from step 1. Iteratively applying Eq. (2.34) L times leads to the final indices (u, v). For each (u, v) we get an ordered list ℓ containing the indices from the original matrices that lead to this index.

3. For each (u, v) calculate the matrix entry

$$m_{u,v} = \prod_{(i,j) \in \ell} a_{i,j}, \tag{2.36}$$

   where $a_{i,j}$ are the entries of the original matrix.

4. If $a_{i,j}$ contains the function f choose the neighbor states based on the assumption and the indices (states) from ℓ of the adjacent matrices.

The corresponding pseudo-code can be found in Algorithm B.1 in Appendix B.

Note that for real data we have to ensure that all CpGs of a given sequence originate from the same cell in order to properly investigate the neighborhood dependencies. Real data rarely covers states of more than a couple of successive CpGs from the same cell with sufficiently deep coverage. Therefore, the

number of contiguous CpGs is usually very limited, such that the explicit construction of the transition matrix for short CpG sequences is feasible in most cases. For a possible larger number of CpGs from advanced measurement techniques we have to resort to more sophisticated methods to obtain the transition matrices or even avoid the generation completely and resort to matrix vector matrix products on the smaller component matrices [18, 19, 20, 32, 118].

Processivity

The detailed mechanisms about the interaction of the Dnmts with the DNA remain elusive. The Dnmts may behave in a processive way, i.e., moving continuously on the DNA strand, or in a distributive manner without directed movement, where attachment and detachment occurs at arbitrary positions on the DNA strand. We therefore would like to test these different assumptions about the methylation mechanisms [44, 61, 87, 102]. A reasonable assumption for Dnmt1 is *processivity from left to right* (assuming 5′ to be at the left and 3′ to be at the right end), due to its link to the replication machinery. The processivity from left to right implies, that a transition already happened at the left neighbor (position $l-1$) but not yet at the right neighbor ($l+1$). This means, given the list of indices $\ell = [\ldots, (i_{l-1}, j_{l-1}), (i_l, j_l), (i_{l+1}, j_{l+1}), \ldots]$ we choose j_{l-1} for the left neighbor state and i_{l+1} for the right neighbor state as an input for the function in step 4 of our algorithmic scheme. Consider for example the transition from a fully unmethylated sequence (∘ ∘ ∘) to a fully methylated sequence (• • •). In this case the correct order of (sub)transitions with their respective probabilities are:

$$\circ\circ\circ \xrightarrow{(2.18)} \bullet\circ\circ \xrightarrow{(2.15)} \bullet\bullet\circ \xrightarrow{(2.21)} \bullet\bullet\bullet$$

Verification

In order to check the correctness of our dedicated implementation for generating matrices with the Kronecker product for L CpGs, we compare the resulting distributions with results from Monte-Carlo (MC) simulations. As initial distribution π_0 we use a discrete uniform distribution which assigns the same probability 4^{-L} to all possible methylation patterns. We then compute the transient distributions $\pi(t)$ after $t = 30$ cell division and subsequent methylation events via

$$\pi(t+1) = \pi(t) \cdot P, \tag{2.37}$$

where P is the total transition matrix, where we assume processivity. Note that $t = 30$ cell divisions is well within the order of cell divisions for biological data [6], where the system still shows a transient behavior. For a larger number of cell

divisions the system may reach a stationary state. However, with the generated transition matrix a stationary analysis is also straightforward.

We perform the corresponding MC simulations of our model with $N = 10^6$ runs to get an independent estimation for the transient pattern probabilities. Note that for that many runs the results from MC simulations are already pretty stable, i.e., the confidence intervals for the estimated transient probabilities are small. In order to not overload the figures, we therefore do not show them here. The distributions for different parameter sets are shown in Fig. 2.9. Panels (a) and (b) show the fully dependent case, where the transition probabilities depend only on the neighbor states and not on the actual maintenance and *de novo* rates μ and τ. In Fig. 2.9 (c) the transition probabilities are completely independent of the neighboring states and depend solely on μ and τ. This case is equivalent to the case where we replace the function f by the respective (constant) transition probabilities. Fig. 2.9 (d) shows a case with some dependency on both neighbor states, where the dependence to the left is slightly stronger. Choosing a wrong transition function in the matrix entries (compared to MC, where it is easier to ensure the correct choice) would affect the distribution in (a) and (b) the most, since there is a full dependency and hence the largest effect from the neighboring states. For the partial dependencies in (d) there should also be an effect if the choices were wrong. In the independent case (c) there can not be a wrong choice, since the transition function is a constant.

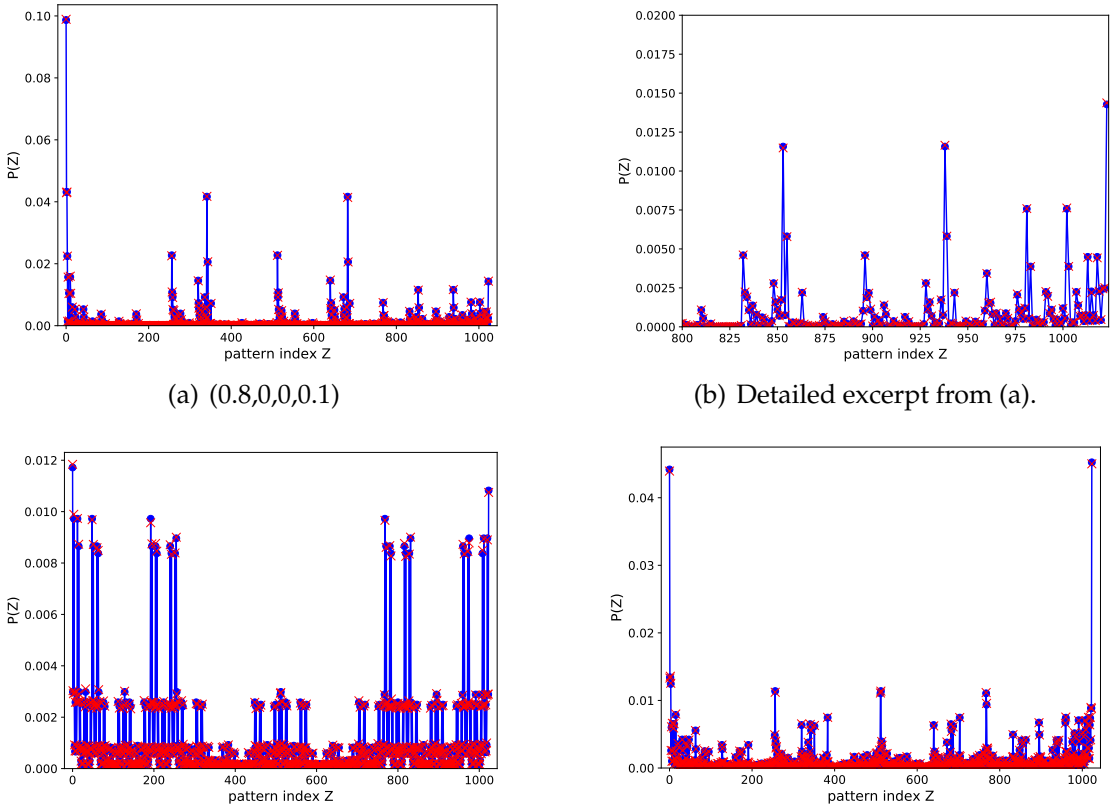
In all cases we observe an almost perfect agreement with only small deviations on some patterns on a very small scale. In order to exclude a flaw in the construction of the transition matrix we compute the Hellinger distance

$$H(P,Q) = \frac{1}{\sqrt{2}} \left(\sum_{i=1}^{k} (\sqrt{p_i} - \sqrt{q_i})^2 \right)^{\frac{1}{2}} \tag{2.38}$$

to compare the similarity of the distributions and to check if the deviations stem from the finite number of MC simulation runs. From Fig. 2.10 it is obvious that with an increasing number of runs the distributions become more and more similar such that we can indeed exclude a flaw in the matrix construction. The small deviations stem from the finite number of runs since for $N = 10^6$ there are still statistical inaccuracies and hence H is quite large (order of 10^{-2}).

Generalizations

The presented SAN framework allows to consider longer CpG sequences and hence the proper investigation of larger genomic regions. The transition matrix for L CpGs can systematically be generated from the small single-CpG matrices and the generation is less prone to errors than an ad hoc approach. It is also



(a) (0.8,0,0,0.1)

(b) Detailed excerpt from (a).

(c) (0.8,1,1,0.1)

(d) (0.8,0.4,0.6,0.1)

Figure 2.9: Comparison of distributions obtained from transition matrices generated from the SAN description (blue dots connected with solid lines) and from MC simulations (red crosses). The parameters for each subfigure are given in the form $(\mu, \psi_L, \psi_R, \tau)$.

pretty easy to adapt the model to test different biological assumptions by using different functions in the transition matrices. In our example, we assumed processivity from left to right, but by changing the functions other assumptions like processivity from right to left (less biologically plausible) or even non-processive (e.g. distributive) behavior can be realized. It is also easily possible to introduce additional reaction parameters for each individual CpG within this framework to generalize the model. Using the same reaction parameters for all CpGs is a strong assumption, especially for the neighborhood dependencies, which should intuitively be different due to the (in general) different distances between CpGs or also due to different base sequences in the DNA (see also Fig. 2.20).

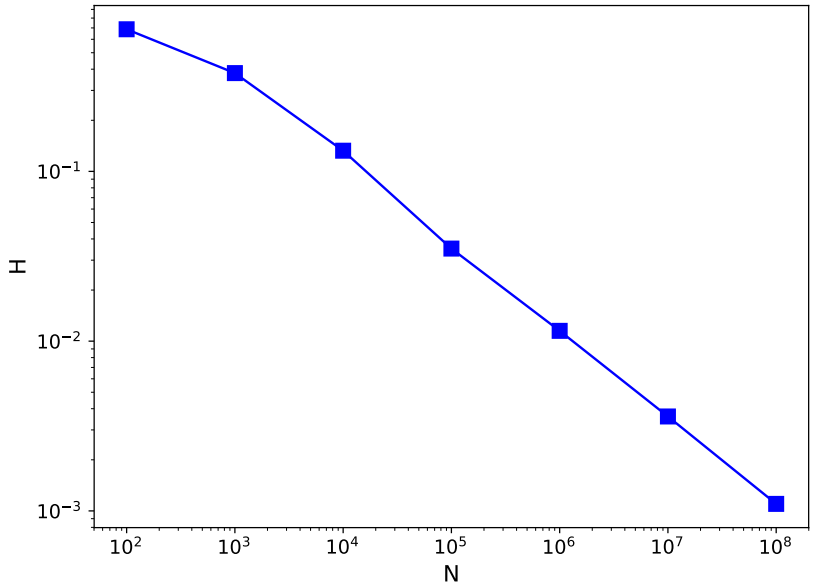


Figure 2.10: Hellinger distance H between distribution obtained from numerical SAN solution and from MC simulations with N runs for the parameter set of Fig. 2.9 (d).

Furthermore, it is also straightforward to apply the SAN approach to more complex methylation models in order to investigate possible neighborhood dependencies. This is especially useful when there are more than four states per CpG as with more states (and hence more possible patterns) the transition matrix grows rapidly (see Fig. 2.11).

In the simplest case, only one DNA strand is considered. Hence there are only two possible states for each CpG, either methylated or unmethylated. In this case even for a quite large number of CpGs the number of patterns remains moderate (Fig. 2.11, blue), however, in this simple case lacks the possibility to include important features, like the distinction of maintenance and *de novo* methylation.

The inclusion of an additional hemimethylated state resolves this issue. However, so far there is no distinction between the two different possible hemimethylated states. Therefore there are only three possible states (Fig. 2.11, yellow). Using hairpin sequencing techniques allows the distinction between the two different hemimethylated states and increases the number of states per CpG to four (Fig. 2.11, green). Since this is the case this thesis focuses on, it will be thoroughly discussed in Section 2.3.7.

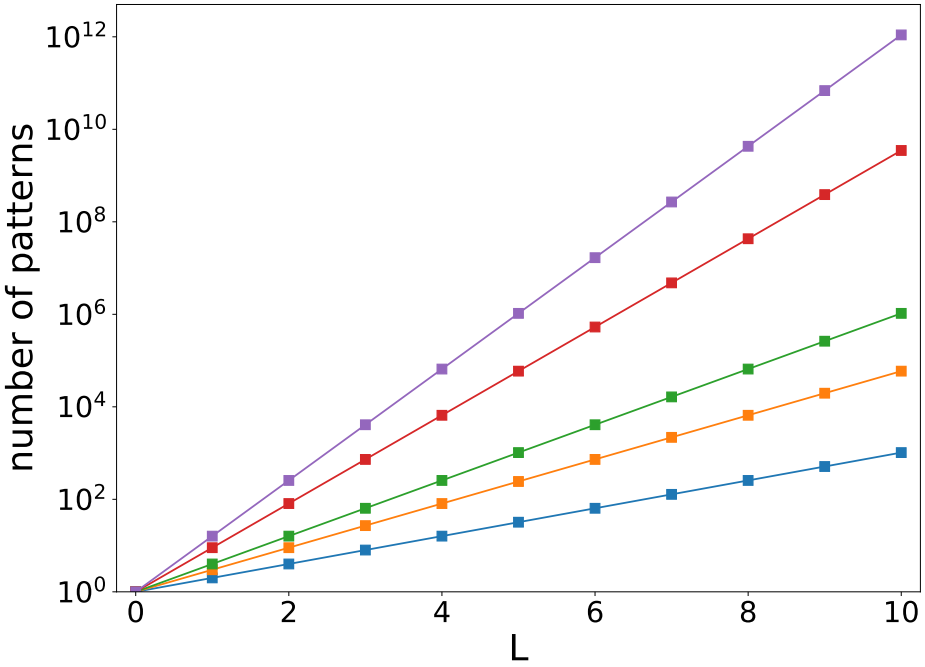


Figure 2.11: Number of patterns for different numbers of CpGs and possible states for a single CpG. The corresponding data (or model) is explained in the main text. Note the log-scale on the Y axis.

Introducing additional hydroxylated Cs as in [38] the number of states per CpG grows from four to nine such that the number of possible patterns grows to 9^L and the size of the transition matrix grows to $9^L \times 9^L$ for L CpGs (Fig. 2.11, red).

With even more possible modifications of C, such as the formylated form 5-formylcytosin, the number of possible states and hence the matrix size grows even more (16^L or $16^L \times 16^L$ respectively; Fig. 2.11, purple). In this case, the SAN description becomes even more useful as it would be very tedious to generate the transition matrix in other ways. It is also possible to apply the SAN approach to continuous time Markov chains or hybrid models as in [77]. Here, the discrete transition matrix was generated with a Kronecker product, while the continuous generator matrix can be generated with a Kronecker sum.

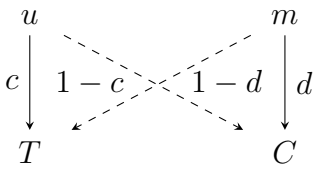


Figure 2.12: Conversions of the unobservable states u, m to observable states T, C with respective rates.

2.3.6 Conversion Errors

The actual methylation state of a C cannot be directly observed. During BS-seq, with high probability every unmethylated C (denoted by u) is converted into thymine (T) and every 5mC (denoted by m) into C. However, conversion errors may occur and we define their probability as $1 - c$ and $1 - d$, respectively, as shown by the dashed arrows in Fig. 2.12. It is reasonable that these conversion errors occur independently and with approximately identical probability at each site and thus the error matrix for a single CpG (containing two Cs) takes the form

$$\Delta_1 = \begin{pmatrix} c^2 & c\bar{c} & c\bar{c} & \bar{c}^2 \\ c\bar{d} & cd & \bar{c}\bar{d} & d\bar{c} \\ c\bar{d} & \bar{c}\bar{d} & cd & d\bar{c} \\ \bar{d}^2 & d\bar{d} & d\bar{d} & d^2 \end{pmatrix}, \tag{2.39}$$

with $\bar{c} = 1 - c$ and $\bar{d} = 1 - d$. Due to the independence of the events this matrix can easily be generalized for systems with $L > 1$ CpGs by recursively using the Kronecker-product

$$\Delta_L = \Delta_1 \otimes \Delta_{L-1} \qquad \text{for } L \geq 2. \tag{2.40}$$

Hence, Δ_L gives the probability of observing a certain sequence of C and T nucleotides for each given unobservable methylation pattern. In order to compute the likelihood $\hat{\pi}$ of the observed BS-seq data, we therefore first compute the transient distribution $\pi(t)$ of the underlying Markov chain at the corresponding time instant[4] t by solving Eq. (2.37) and then multiply the distribution of the unobservable patterns with the error matrix.

$$\hat{\pi} = \pi(t) \cdot \Delta_L. \tag{2.41}$$

Note that this yields a hidden Markov model with emission probabilities Δ_L. In the following the values for c were chosen according to [6]. Since the value

[4]The number of cell divisions is estimated from the time of the measurement since these cells divide once every 24 hours.

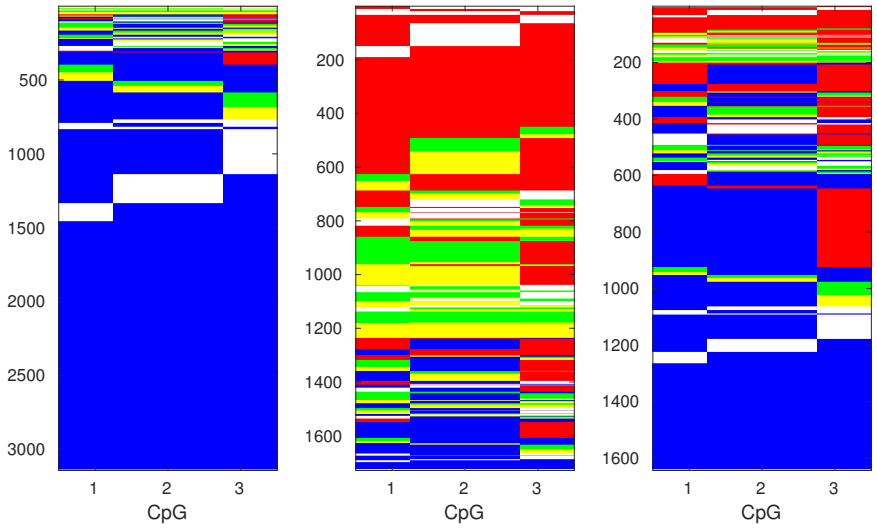


Figure 2.13: Representations of WT (left), Dnmt1KO (middle) and Dnmt3a/b DKO (right) data for mSat. On the X axis the CpGs and on the Y axis the measured cells are shown. The different colors encode the states as follows: Red: 0, green: 1, yellow: 2, blue: 3, and white: "no measurement".

for d was not determined in [6], we measured the conversion rate $d = 0.94$ in an independent experiment under comparable conditions [38]. In this study we used hairpin linkers, which contain C, 5mC, as well as 5hmC. After sequencing we determine the conversion state of each particular C from within each read. Note, that we calculated the average conversion rate of all experiments for the present study.

2.3.7 Data

For our analysis we focused on hairpin data of the single copy genes Afp (5 CpGs) and Tex13 (10 CpGs) as well as the repetitive elements IAP (intracisternal A particle) (6 CpGs), L1 (Long interspersed nuclear elements) (7 CpGs) and mSat (major satellite) (3 CpGs). During the workflow of hairpin bisulfite sequencing, the two DNA strands are linked together covalently, i.e. the methylation status of both strands from an individual chromosome (DNA molecule) is known. Repetitive elements occur in multiple copies and are dispersed over the entire genome. Therefore they allow capturing an averaged, more general behavior of methylation dynamics.

The statistics for all data sets are summarized in Tab. 2.2. Note that only for mSat we have more samples than possible patterns when considering the

Table 2.2: Number of CpGs L and sample sizes N for the different data sets.

Locus	L	N_{1KO}	N_{3abKO}	N_{WT}
mSat	3	1 191	1 187	2 285
Afp	5	134	186	564
IAP	6	182	457	609
L1	7	174	128	273
Tex13	10	394	601	1 044

whole sequence of all CpGs. Amongst others, this is one of the reasons why we restrict the discussion mostly to sequences of three CpGs. The mSat data sets are visualized in Fig. 2.13. Note that the WT data is almost always fully methylated, while the Dnmt1KO data is mostly un- or hemimethylated. The Dnmt3a/b DKO data is somewhat in between. Fully methylated CpGs are displayed in blue, hemimethylated CpGs in yellow and green, depending on which C is methylated, and unmethylated CpGs in red. White indicates that there is no measurement for the respective CpG. In this case the sequence has to be discarded for further analysis, because we need a pattern of at least length 3 for our model. For analyzing loci with more than three CpGs all partial sequences with three adjacent CpGs can be kept. Note that we can only use partial sequences of the same CpGs, because different CpGs may in general show different behaviors. If we, for example, analyze a locus with four CpGs and for a certain measurement we have no data for the first CpG, we can still keep the pattern from CpGs 2-4 if we would like to analyze those three CpGs. For investigating CpGs 1-3 or all four CpGs simultaneously, the respective measurement has to be discarded due to the missing data for the first CpG.

Therefore, the actual sample size is smaller than the number of measured cells, e.g., we have 1 729 measured cells in the Dnmt1KO data set for mSat (Fig. 2.13 middle), but only 1 191 of them are usable (see Tab. 2.2). For the same reason the sample sizes for partial sequences may be larger than the numbers for the whole sequence given in Tab. 2.2. For example, in the L1 Dnmt1KO data set there are only 174 usable measurements for the whole sequence of seven CpGs, but 1 047 measurements that can be used, when only considering the first three CpGs.

2.4 Parameter Estimation Methods

In order to estimate the parameters $\theta = (\mu, \psi_L, \psi_R, \tau) \in [0,1]^4$, there are several possible methods available. Here we discuss (and apply) three examples of conceptually different parameter estimation techniques: a moment-based approach (Generalized Method of Moments), a Bayesian approach (Approximate Bayesian Computation), and a likelihood-based approach (Maximum Likelihood Estimator).

2.4.1 Generalized Method of Moments

We start of with a moment-based parameter estimation method. To ensure identification of the parameters, we use the following quantities, which are based on the methylation state and independent of the labeling of these states. Since the labeling of states in our model might be chosen arbitrarily, i.e., there is no associated interpretation of the state labels, such as number of proteins later in Chapter 3, the classical moments (such as the mean state) are meaningless here. Instead, we define a set of more suitable moments.

We consider a pattern of L CpGs in the k-th measured cell, $k \in \{1, \dots, N\}$. Let $M_i^{(k)} \in \{0,1\}$ be the methylation state of the upper C in CpG i, where 1 represents a methylated and 0 an unmethylated C. Let $S_i^{(k)} \in \{0,1,2\}$ be the number of methylated Cs of CpG $i \in \{1, \dots, L\}$. We consider moments of the following RVs:

- (the horizontal average of) the methylation level on the upper strand

$$X_k = \frac{1}{L} \sum_{i=1}^{L} M_i^{(k)}, \tag{2.42}$$

- the squared difference of the methylation level and the (cell population) average of the methylation level

$$(X_k - \bar{X})^2, \tag{2.43}$$

- a quantity to measure the fraction of consecutive methylated Cs on the upper strand

$$\frac{1}{L-1} \sum_{i=1}^{L-1} \left(M_i^{(k)} \cdot M_{i+1}^{(k)} \right), \tag{2.44}$$

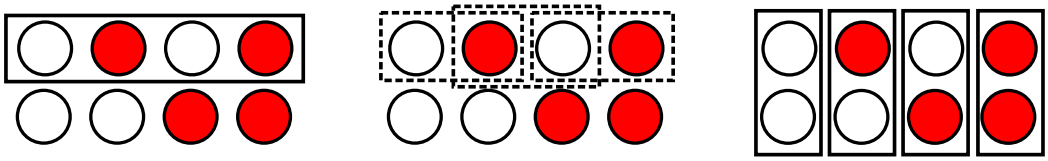


Figure 2.14: Visual representation of the RVs (2.42) (left), (2.44), (2.45) (middle), and (2.46) (right) for 4 CpGs and example pattern 0123. When only considering the upper strand, this pattern is converted to 0101. (2.43) and (2.47) correspond to the variances of (2.42) and (2.46).

- a quantity to measure the fraction of consecutive unmethylated Cs on the upper strand

$$\frac{1}{L-1}\sum_{i=1}^{L-1}\left((1-M_i^{(k)})\cdot(1-M_{i+1}^{(k)})\right), \tag{2.45}$$

- the number of methylated Cs for each CpG

$$S_i^{(k)}, \tag{2.46}$$

- the squared difference of the number of methylated Cs and the cell population average of methylated Cs in each CpG

$$(S_i^{(k)}-\bar{S}_i)^2. \tag{2.47}$$

Note that since our model is strand symmetric, the upper and lower strand behave equivalently and the moments based on Eqs. (2.42)-(2.45) are identical for both strands. Therefore, w.l.o.g. we consider only the quantities for the upper strand. A visual representation of the quantities can be found in Fig. 2.14. Further, note that all these quantities only take small values due to their definition and all have the same (or a very similar) order of magnitude.

We selected the above quantities based on some considerations: The methylation level (2.42) and number of methylated Cs for each CpG (2.46) are obvious choices. The squared differences to their average (2.43) and (2.47) are later needed to obtain variances. Since the model contains neighborhood dependencies, i.e., the state of one CpG may influence (or even determine) the states of its neighbors, the number of consecutive (un)methylated Cs (2.44) and (2.45) contain valuable information. Note that it is not possible to distinguish between alternating states and consecutive opposite states with only one of these quantities. For example, with Eq. (2.44) only, it is impossible to distinguish the

patterns 00000 and 10101 ($L = 5$). The combination of (2.44) and (2.45) contains this information. We will later investigate which of the defined quantities (2.42)-(2.47) are mandatory for the successful parameter identification and estimation.

For each measured cell k, we collect the quantities defined in Eqs. (2.42)-(2.47) (or a subset thereof) in a random vector $\mathbf{Y}_k$. For L CpGs, each $\mathbf{Y}_k$ has (depending on how many moments are used, up to) $m = 4 + 2L$ entries. The corresponding sample moments are denoted by

$$\bar{\mathbf{Y}} = \frac{1}{N} \sum_{k=1}^{N} \mathbf{Y}_k \tag{2.48}$$

and the theoretical moments, which can either be obtained from the numerical solution of the model or from MC simulations, are denoted by $\mathbf{m}(\theta)$, where θ is the vector of model parameters. We then define the cost functions as

$$g_r(\theta) := \bar{Y}^{(r)} - m_r(\theta), \tag{2.49}$$

where r denotes the r-th entry in the corresponding vectors.

An obvious inference approach would be to consider the ordinary least squares estimator

$$\hat{\theta} \;=\; \arg\min_{\theta} \sum_{r=1}^{m} \left(g_r(\theta)\right)^2, \tag{2.50}$$

where m is the number of moment constraints. Under certain conditions related to the identification of the parameters as discussed below, this estimator is consistent (converges in probability to the true value of θ) and asymptotically normal. However, its variance may be very high. This problem can be mitigated by choosing appropriate weights for the summands in (2.50). Moreover, since correlations between the cost functions $g_r(\theta)$ exist, a more general approach that considers mixed terms is needed. This leads to a class of estimators, called generalized method of moments (GMM) estimators that have been introduced by Hansen [52]. The idea is to define the estimator as

$$\hat{\theta} \;=\; \arg\min_{\theta} \mathbf{g}(\theta)' W \mathbf{g}(\theta), \tag{2.51}$$

where $\mathbf{g}(\theta)$ is the column vector with entries $g_r(\theta)$, $r = 1, \ldots, m$, and W is a positive semi-definite weighting matrix. Note that by defining $f_r(Y, \theta) = Y^{(r)} - m_r(\theta)$ we see that

$$g_r(\theta) = \frac{1}{N} \sum_{k} f_r(Y_k, \theta) = \frac{1}{N} \sum_{k} Y_k^{(r)} - m_r(\theta) \tag{2.52}$$

is the sample counterpart of the expectation $E[f_r(Y,\theta)]$. The latter satisfies

$$\theta_0 = \arg\min_{\theta} E[\mathbf{f}(Y,\theta)]' W E[\mathbf{f}(Y,\theta)], \tag{2.53}$$

where $\mathbf{f}(Y,\theta)$ is the column vector with entries $f_r(Y,\theta)$ and θ_0 is the true value of θ. Note that the choice $W = I_m$, where I_m is the $m \times m$ identity matrix, gives the least-squares estimator (2.50) with m terms while for general W there are $\frac{m\cdot(m+1)}{2}$ terms in the objective function (with m being the dimension of $\mathbf{g}(\theta)$). In addition, we remark that in general W may depend on θ and/or the samples Y_k.

Here we assume that identification of θ is possible, i.e., we require that the number of the moment constraints used is at least as large as the number of unknown parameters and

$$E[\mathbf{f}(Y,\theta)] = \mathbf{0} \text{ if and only if } \theta = \theta_0. \tag{2.54}$$

Moreover, the theoretical moments $m_r(\theta)$ should not be functionally dependent (see Chapter 3.3 in [49]) to ensure that the information contained in the moment conditions is sufficient for successfully identifying the parameters.

By applying the central limit theorem to the sample moments, it is possible to show that the GMM estimator is consistent and asymptotically normally distributed and that its variance becomes asymptotically minimal if the matrix W is chosen such that it is proportional to the inverse of the covariances between the $Y_k^{(r)}$ [52]. Intuitively, whenever a sample moment has high variance, its weight is decreased compared to sample moments with lower variance. Formally, we define $\mathbf{Y}_k$ as the random vector with entries $(Y_k)^{(r)}$ for $r = 1,\ldots,m$ and omit the subindex k if it is not relevant. Then,

$$F(\theta_0) = COV[\mathbf{Y},\mathbf{Y}] = E[\mathbf{f}(Y,\theta_0)\mathbf{f}(Y,\theta_0)^T] \tag{2.55}$$

and choosing $W \propto F^{-1}$ will give an estimator with smallest possible variance, i.e., it is asymptotically efficient in this class of estimators [49, 52]. Note that the covariance depends on the (unknown) real parameter value θ_0. Using an estimated value $\tilde{\theta}$ instead of the true one may lead to "misspecification", i.e.,

$$E[\mathbf{Y}] \neq \mathbf{m}(\tilde{\theta}). \tag{2.56}$$

In this case, the above estimator is no longer consistent and the weight matrix W might be suboptimal.

An estimator for F that is consistent is then given by [49]

$$\hat{F} = \frac{1}{N}\sum_{k=1}^{N}(\mathbf{Y}_k - \overline{\mathbf{Y}})(\mathbf{Y}_k - \overline{\mathbf{Y}})^T, \tag{2.57}$$

where $\overline{\mathbf{Y}}$ is the vector containing the sample moments defined in Eq. (2.48). Note that Eq. 2.57 is the sample counterpart of Eq. (2.55). It is important to note, that instead of the theoretical moments, the sample means are subtracted here. In the sequel, we refer to the estimator based on Eq. (2.57) as the *demean estimator*. This estimator removes the inconsistencies in the covariance matrices estimated from the sample moments by "demeaning".

Later, in Section 3.2.4, we will discuss more methods to obtain an ideal weight matrix and show how to adapt the GMM to multiple species and time points. For the remainder of this chapter, we will focus on the demean estimator with data from one time point only.

2.4.2 Approximate Bayesian Computation

According to Bayes' theorem

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)} \tag{2.58}$$

the conditional density of parameter θ given data D (posterior) is determined by the conditional density of D given θ (likelihood), the prior $p(\theta)$ and the evidence $p(D)$. Note that the prior $p(\theta)$ is independent of D and is therefore used to incorporate initial beliefs about the parameter values. If nothing is known a uniform prior is usually an appropriate choice. In general, posterior and prior stem from different families of distributions.

Since usually it is computationally too expensive (as for large L in our model) or even infeasible to evaluate the likelihood, one has to resort to methods which circumvent the likelihood evaluation. One of these methods is the approximate Bayesian computation (ABC). where the real data D is compared to data $\tilde{D}$ generated from the model for different parameters θ distributed according to the prior $p(\theta)$. The results are then compared via some suitable measure $d(D, \tilde{D})$. If $d(D, \tilde{D}) \leq \varepsilon$, where $\varepsilon \geq 0$ is a small tolerance, the respective parameter is added to the posterior and otherwise discarded. Note that it is also possible to not compare the data directly and rather use summary statistics $S(D)$ and $S(\tilde{D})$. For sufficient summary statistics there is no additional error introduced by using the statistic.

In general, the rejection rate in a simple ABC algorithm is quite high. We therefore resort to a variant of ABC called ABC-sequential-Monte-Carlo (ABC-SMC) algorithm. The main idea is to first roughly scan the parameter space and keep the best results as a first suggestion for the posterior. Then these results are improved by slowly decreasing the tolerance ε and search preferentially in the parts of the parameter space that yielded better results.

We now describe the simple ABC-SMC version that we used in more detail: As data D we use the distribution over the patterns and obtain the respective distributions $\tilde{D}$ from the model either by MC simulations or the numerical solution of Eq. (2.37) (only feasible for small/moderate L). Since initially nothing is known about the parameters, except that they are probabilities, i.e. take on values between 0 and 1, we chose a uniform prior, i.e. $p(\theta) \sim U(0,1)$ for all four parameters. We then calculate $d(D,\tilde{D})$ for N_1 samples and keep the n best results. Note that it is usually preferable to specify the size of the posterior rather than ad hoc choosing a tolerance. The value of the distance function depends on a variety of factors, amongst others on the distance function itself. It is therefore hard to determine only by considering the value of the distance function, whether some parameters are good or bad and hence what is a good choice for the acceptance threshold. A too large threshold leads to a too wide posterior and a too narrow threshold leads to too many rejections and hence to long run times due to the large number of required runs. It is therefore preferable too specify the size of the posterior and adaptively calculate and update the threshold.

From the accepted parameter values, we then have a list V that contains the parameter values and the corresponding distances. For the measure d suitable choices are for example the Euclidean distance or the Hellinger distance (Eq. (2.38)). Based on the distances in V we calculate the tolerance ε as the average distance

$$\varepsilon = \frac{1}{n} \sum_{d_i \in V} d_i \tag{2.59}$$

and weights for each parameter set in the list

$$w_j = \left(d_j \sum_{d_i \in V} d_i^{-1} \right)^{-1} . \tag{2.60}$$

The weights are used to search preferentially around good parameter values (small distance, large weight) and to a lesser extend around worse parameter values (big distance, small weight). After randomly choosing parameter values from V according to the weights, these parameters are used a means for a multivariate normal distribution with variance σ. Note that there are sophisticated methods for an optimal choice of σ. Here, we chose a small constant σ which should neither be too small, since then the parameter space is not sufficiently explored, nor too large since then the rejection rate becomes too large. We then sample new parameter values from this normal distribution. Note that the normal distribution may yield parameter values outside of the interval between 0 and 1. In this case these parameters are discarded immediately. Otherwise

the distance is calculated and if it is smaller than the current tolerance ε the corresponding parameters and distance replace the entry in V with the largest distance, i.e. the number of entries in V and ultimately in the posterior remains constant. We then recompute the tolerance and weights and repeat this process until some criterion is met. Some possible criteria are a maximum number of tries N_2, a minimum of change in the tolerance, or a small standard deviation of the posterior. The final values of the parameters in V form the posterior. The parameter estimations can be extracted from the posterior, for example by taking the mean or the median. The pseudo-code for this simple ABC-SMC version can be found in Algorithm B.2 in Appendix B.

2.4.3 Maximum Likelihood Estimator

A commonly used parameter estimation technique is the Maximum Likelihood Estimator (MLE). Given a set of observed patterns $x_1, \ldots, x_N$ of L CpGs and the parameterized pattern distribution $\hat{\pi}(\theta)$ from the numerical solution of (2.37) and (2.41) for a given time t, we can derive the likelihood

$$\mathcal{L}(\theta) = \prod_{k=1}^{N} \hat{\pi}_\theta(x_k), \tag{2.61}$$

where $\hat{\pi}_\theta(x_k)$ is the probability of pattern x_k from the numerical solution, given parameters θ. Note that in order to write the likelihood in this product form, $x_1, \ldots, x_N$ have to be realizations of independent and identically distributed random variables, which is the case for the independently observed patterns. Since there are only 4^L possible patterns for L CpGs, we can rewrite Eq. (2.61) into

$$\mathcal{L}(\theta) = \prod_{j=1}^{4^L} \hat{\pi}_j(\theta)^{N_j}, \tag{2.62}$$

where N_j is the number of occurrences of pattern j in the observed data. Note that

$$N = \sum_{j=1}^{4^L} N_j. \tag{2.63}$$

In practice usually the log-likelihood

$$\ell(\theta) = \log(\mathcal{L}(\theta)) = \sum_{j=1}^{4^L} \log(\hat{\pi}_j(\theta)) \cdot N_j \tag{2.64}$$

is maximized instead of the likelihood. Since the logarithm is a monotonic function the maximum occurs at the same value of $\hat{\theta}$ for both $\mathcal{L}$ and ℓ. Applying the logarithm has the advantage that it transforms the numerically unstable product into a sum, which also has asymptotic properties that are easier to analyze. The parameters $\theta = \hat{\theta}$ are then chosen in such a way that ℓ is maximized, i.e.,

$$\hat{\theta} = \arg\max_{\theta} \ell(\theta). \tag{2.65}$$

In order to ensure that the global maximum in $[0, 1]^4$ is found during the optimization, we ran the estimation several times with different random starting points. In all cases the estimation yielded the same results, such that it is very likely that indeed the global optimum was found.

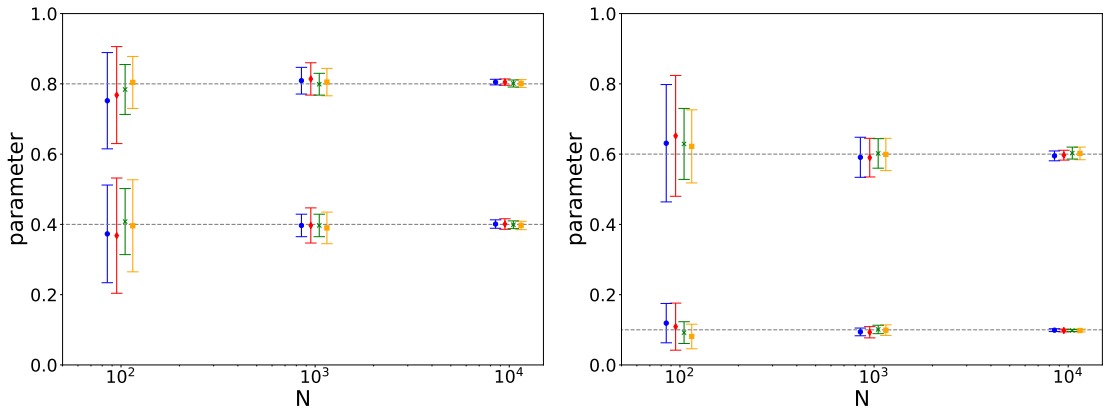
2.5 Results

We now present the results for our model. First, we test the different parameter estimation methods and then proceed with the method best suited for our case. Note that, except for testing the different parameter estimation methods, we restrict the discussion of the results to 3 CpGs, mainly due to two reasons: Although it is possible (especially with the presented SAN framework) to generalize the model to more CpGs, the assumption of the same parameters for all CpGs becomes more and more unrealistic. In principle one could introduce separate parameters for each CpG, however, together with the increasing model complexity due to the increasing number of possible states, the (currently) available biological data is usually not enough. Typical hairpin data sets contain information of about 100-1 000 cells with connected CpGs which is not enough to analyze models with many states and parameters.

2.5.1 Parameter Estimation

Here, we test the different parameter methods on artificial data, i.e., we know the real parameters and can therefore assess the qualtity of the prediction. For GMM and MLE, which are both significantly faster than ABC, we also perform estimations on real biological data.

If not stated otherwise all estimations are performed for model 1, i.e. processive maintenance first and processive *de novo* methylation afterwards, where the direction is assumed to be from the left to the right.



(a) True parameters: $\mu = 0.8$, $\psi_L = 0.4$

(b) True parameters: $\tau = 0.1$, $\psi_R = 0.6$

Figure 2.15: Mean and standard deviation of the estimated parameters $\hat{\theta}_{\text{GMM}}$ and $\hat{\theta}_{\text{MLE}}$ from 25 estimations for MC simulation data with a sample size of N. The red (orange) bars show the GMM estimations for 3 (4) CpGs and the blue (green) bars the MLE estimations for 3 (4) CpGs.

Generalized Method of Moments

In order to determine the accuracy of the GMM approach applied to parameters of spatial methylation models, we initially use artificial data (with known parameters) generated from Monte-Carlo (MC) simulations. The known parameters enable us to ensure that the estimations indeed yield the correct parameters. Additionally, we compare the GMM estimations to results from a MLE (Eq. (2.65)), in order to compare their performances in terms of accuracy and requirements of the available data. For each parameter set and sample size we generate 25 data sets from MC simulations and use them to obtain the mean and standard deviations for the estimates.

In Fig. 2.15 we plot the results for parameters $\theta = (\mu, \psi_L, \psi_R, \tau) = (0.8, 0.4, 0.6, 0.1)$, where the red (orange) bars show the GMM estimations for $L = 3$ ($L = 4$) CpGs and the blue (green) bars the MLE estimations for $L = 3$ ($L = 4$) CpGs for different sample sizes N, respectively. Note that we assume identical parameters for all CpGs of the pattern and N is the number of single-cell pattern samples at the selected position. Also note that the bars have a little offset to the left/right of the actual sample size in order to increase the clarity of the presentation. We observe that both GMM and MLE show a very similar performance in terms of accuracy for all four parameters. Furthermore, a relatively modest sample size of 100-$1\,000$ is already enough to obtain reliable estimates.

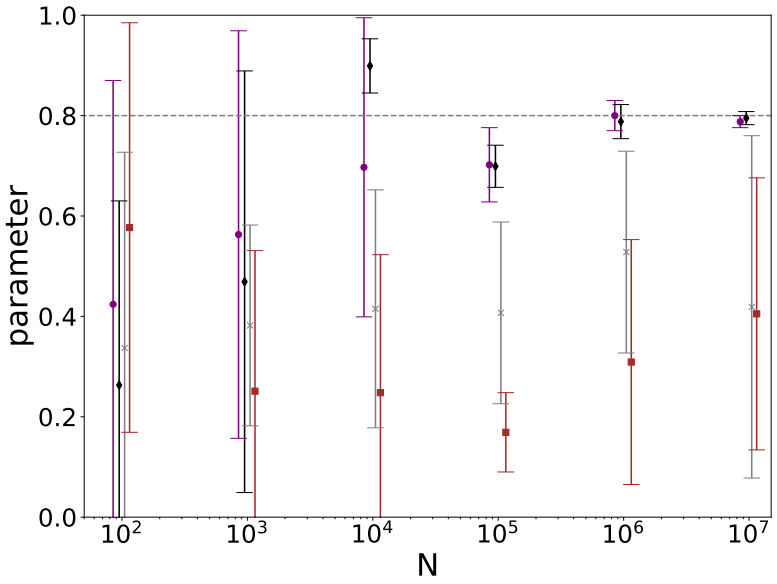


Figure 2.16: Estimations for μ for different subsets of moments. Purple: (2.44), (2.46); black: (2.42), (2.44), (2.45), (2.46); gray: (2.42), (2.44), (2.45); brown: (2.42), (2.46)

Note that not all moments derived from Eqs. (2.42)-(2.47) are needed to ensure identifiability of the parameters. In Fig. 2.16, we plot results for different subsets of moments. Without the variances (Fig. 2.16, black bars, moments of Eqs. (2.42), (2.44), (2.45), (2.46)) and additionally even without the methylation level and the successive unmethylated CpGs (Fig. 2.16, purple bars, moments of Eqs. (2.44), (2.46)) the parameters can still be estimated correctly, however, only with significantly larger sample sizes. On the other hand, when we only consider the methylation level and the number of methylated Cs per CpG (Fig. 2.16, brown bars, moments of Eqs. (2.42), (2.46)) or the methylation level and the successive (un)methylated CpGs (Fig. 2.16, gray bars, moments of Eqs. (2.42), (2.44), (2.45)) the GMM can not estimate the real parameters, even for very large sample sizes. Fig. 2.16 shows the estimation only for μ, however, the results are very similar for the other parameters and are therefore not shown. Hence, at least one of the moments derived from the number of successive (un)methylated CpGs (Eq. (2.44) or (2.45)) as well as the number of methylated Cs per CpG (Eq. (2.46)) are needed to ensure identification of the parameters.

Intuitively, the reason that these moments contain enough information to successfully identify the parameters is that due to the neighborhood dependen-

cies, the average number of consecutive (un)methylated CpGs is a good indicator for the strength of the neighborhood dependence. Furthermore, since each CpG is influenced by its neighboring CpGs, each CpG in general may have a different average number of methylated Cs. The other moments are less informative. The average methylation level in Eq. (2.42), for example, gives no hint about the distribution of methylation, i.e. if it is spread uniformly over all CpGs or only concentrates on certain areas. On the other hand, once identification is ensured, additional information from such moments helps to estimate the parameters more accurately for smaller sample sizes. For 100–$1\,000$ sample patterns, which is the order of magnitude for the hairpin bisulfite sequencing data considered later, all moments should be considered to achieve an accurate estimation.

We also perform estimations for different parameter sets with stronger/weaker dependencies, higher/lower methylation efficiencies and combinations thereof. The results are in agreement with the results in Fig. 2.15 and 2.16, i.e., GMM and MLE show a similar accuracy if the sample size is at least of the order of hundreds and also the moment subsets comparison gives very similar results. We therefore do not present detailed results for these parameter sets.

Finally, we apply the GMM to the hairpin bisulfite sequencing data sets from mouse embryonic stem cells in [6]. During hairpin bisulfite sequencing, the two DNA strands are linked together covalently such that the methylation status of both strands can be measured simultaneously [80]. Our data sets consist of data for single copy genes, which occur only once in the genome, as well as repetitive elements, which occur in multiple copies over the whole genome. For single copy genes, we have data for Afp (5 CpGs) and Tex13 (10 CpGs). For the repetitive elements, the data stems from IAP (intracisternal A particle; 6 CpGs), L1 (Long interspersed nuclear elements; 7 CpGs) and mSat (major satellite; 3 CpGs). We focus on Dnmt1KO data, i.e. only Dnmt 3a/b is active, since previous findings suggest, that in general only Dnmt 3a/b shows a dependence on the left neighbor, while Dnmt1 acts independent of the neighborhood [86].

Since the number of possible states grows exponentially with the number of CpGs, i.e. for L CpGs there are 4^L possible states, the numerical solution is no longer feasible, due to large memory requirements for more than 5 CpGs. We therefore estimate the theoretical moments via MC sampling of the model. Due to finite size effects and statistical inaccuracies these moments are not exact anymore. In order to have an estimate for these variations we compute the confidence interval

$$\bar{m}_q \pm 1.96 \cdot \sqrt{\frac{S_q^2}{N}}, \tag{2.66}$$

where 1.96 is the approximate value of the corresponding percentile point of the normal distribution for a confidence level of 95%, $\bar{m}$ and S^2 are the sample mean and variance of the quantities in Eqs. (2.42)-(2.47) for a sample size of N. We find that for $N = 1\,000$ the relative width of the confidence interval is ≤ 0.1 for all moments and parameter sets and use this sample size for the approximation of the theoretical moments.

Since we have only one data set for each locus available, we use bootstrapping to generate 25 samples and again calculate the mean and standard deviations of the estimators. The results for all available loci are summarized in Tab. 2.3. Note that the standard deviations are rather large due to multiple reasons. First of all, the aforementioned variability in the (MC sampled) theoretical moments leads to a variability in the estimates as well. Furthermore, we use the same parameters for all CpGs. Hence, the results represent the average dependency and methylation efficiency at this position (spanning several CpGs). Introducing separate parameters for each CpG results in $4L$ parameters and may lead to identifiability problems, due to the in general low coverage. For the artificial data considered above, we used the same parameters to generate the data, such that the parameters for each CpGs were indeed identical in this case. Finally, the number of pattern samples that can be considered for the estimation is often very small when considering all CpGs, since often the methylation state for one (or more) of the CpGs is missing, such that we have to omit the whole measurement (see Tab. 2.2 for detailed numbers). Nevertheless, the results are in good agreement with results from the other estimation methods presented later, i.e., for Dnmt 3a/b there is, in general, only a dependence on the left neighbor.

Although the model's moments can be estimated by MC sampling, a numerical approach to compute the moments without calculating the full underlying distribution is desirable. As future work, we plan to derive moment equations that allow a fast numerical computation of the statistical moments. This would allow to obtain accurate estimates very efficiently also in the case of longer methylation patterns and to estimate parameters on a whole-genome scale.

Approximate Bayesian Computation

Again, we investigate the accuracy of the parameter estimation by using artificial data from MC simulations with known parameters $\theta = (\mu, \psi_L, \psi_R, \tau) = (0.8, 0.4, 0.6, 0.1)$ for different sample sizes. We performed the parameter estimation with the simple ABC-SMC method presented in Algorithm B.2. For the first step in the algorithm, i.e. the rough scanning of the parameter space, we draw $N_1 = 10\,000$ random parameter sets from a uniform distribution. In the second step, i.e. the improvement of the initial proposal for the posterior, which

Table 2.3: Mean and standard deviations for GMM for BS-seq hairpin data (Dnmt1KO) from different loci, obtained from 25 bootstrap samples. The number of CpGs and the sample sizes can be found in Tab. 2.2.

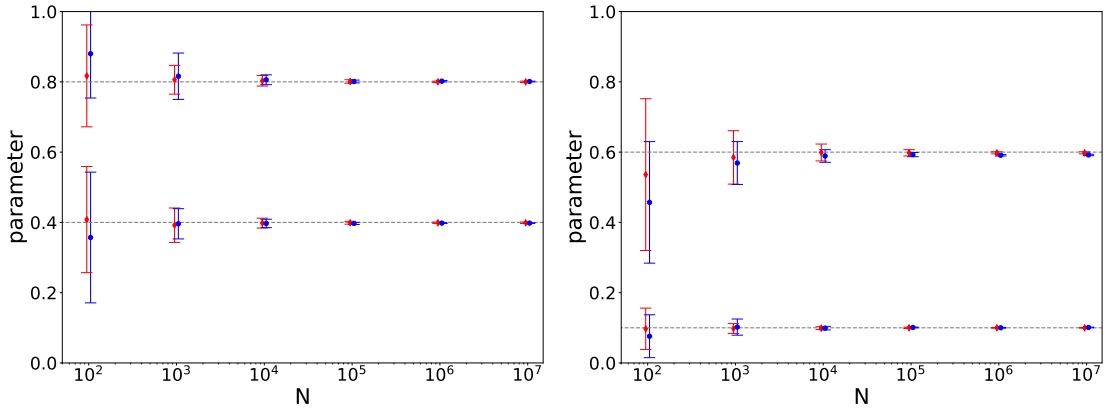
Locus	μ	ψ_L	ψ_R	τ
mSat	0.3278 ± 0.1836	0.2388 ± 0.1784	0.9624 ± 0.0743	0.0069 ± 0.0157
Afp	0.3700 ± 0.3254	0.4357 ± 0.3126	0.5254 ± 0.2833	0.4745 ± 0.2598
IAP	0.5736 ± 0.1611	0.3868 ± 0.2738	0.9388 ± 0.1044	0.0264 ± 0.0356
L1	0.6147 ± 0.2751	0.9443 ± 0.1968	0.9596 ± 0.1959	0.0401 ± 0.1720
Tex13	0.7039 ± 0.3474	0.5990 ± 0.3753	0.9688 ± 0.0709	0.9626 ± 0.0984

consists of the n best results from step 1, we draw $N_2 = 100\,000$ random parameters from a normal distribution and replace the worst performing parameters in the posterior by the new ones, if the distance is smaller than the threshold calculated from Eq. (2.59). The means are the current parameters and we fix the standard deviation to $\sigma = 0.01$ for all four parameters. As a distance function we use the Euclidean distance here.

For each parameter set and sample size N we generate 25 data sets from MC simulations and perform the parameter estimation. We also test different sizes of the posterior n. As an estimator we choose the mean of the posterior and show the corresponding sample mean and sample standard deviation from the 25 data sets in Fig. 2.17. Note that the shown standard deviations only stem from the variety in the data, i.e. from the different means from the 25 approximated posteriors and therefore only form a lower bound. We choose this approach to enable the direct comparison with other methods, where the presented standard deviations of the estimations also only stem from the variety in the data. The standard deviations of the posteriors are therefore not included here and will be discussed later.

We compare the results for posteriors of size $n = 100$ (red) and $n = 1\,000$ (blue) in Fig. 2.17. For both sizes the estimations show a similar accuracy, with a slight advantage for $n = 100$ for very small sample sizes. For larger sample sizes the results are practically indistinguishable. In terms of standard deviations the results are also very similar for both posterior sizes.

As for the posteriors themselves, we show the resulting posterior for an artificial data set of size $N = 10^6$ for a different N_2, i.e., the number of steps to improve the approximation in the ABC algorithm, for the maintenance probability μ in Fig. 2.18. The other parameters show a similar behavior and are therefore not shown here. We explicitly show the resulting histogram from the



(a) True parameters: $\mu = 0.8$, $\psi_L = 0.4$ (b) True parameters: $\tau = 0.1$, $\psi_R = 0.6$

Figure 2.17: Mean and standard deviation of the estimated parameters $\hat{\theta}_{\text{ABC}}$ from 25 estimations for MC simulation data with a sample size of N and 3 CpGs. The red bars show the ABC estimations for a posterior of size 100, the blue bars for a posterior of size 1 000.

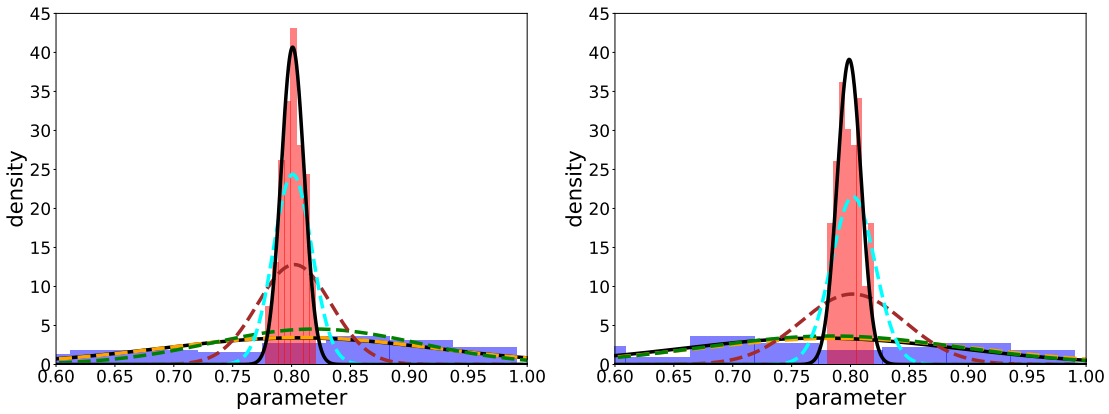
accepted values in the posterior for $N_2 = 0$ (blue), which corresponds to the initial proposal of the posterior after the rough scanning of the parameter space, and the posterior after $N_2 = 10^6$ steps (red) with their respective Gaussian fits (solid black lines). For the values of N_2 in between, we only show the resulting Gaussian fit (dashed lines). The color scheme and the fitting parameters can be found in Tab. 2.4. Note that we use these means as the estimation for the model parameters. Also note, that for the same N_2 the posterior of smaller size is sharper, i.e., has the smaller standard deviation (sd).

This intuitively makes sense, since there are more steps necessary to update the larger posterior. Furthermore, due to the larger posterior size the threshold is more stable under changes of single values and hence decreases slower, such that it is more likely that non-optimal values will still be accepted. On the other hand, the larger posterior allows for more exploration of the parameter space.

Finally, note that different parameter sets as well as different distance functions yield very similar results in terms of performance and accuracy.

Maximum Likelihood Estimator

Since the results for artificial data for the MLE have already been presented in the GMM section, we focus here on the hairpin data for the single copy genes and repetitive elements as introduced in the Section 2.3.7. If a locus contains more than three CpGs, the analysis is done for all sets of three adjacent sites



(a) Posterior size $n = 100$ (b) Posterior size $n = 1\,000$

Figure 2.18: Example posteriors for different sizes n for $N_2 = 0$ (blue) and $N_2 = 10^6$ (red) with Gaussian fits for the posteriors with $N_2 = 0, 10^2, 10^3, 10^4, 10^5$ and 10^6 for the (true) maintenance probability $\mu = 0.8$. The coloring scheme as well as the fitting parameters for the Gaussian fits can be found in Tab. 2.4.

independently, in order to keep computation times short and memory requirements low. In the sequel, we mainly focus on the estimated dependence parameters ψ_L and ψ_R and on the prediction quality of the different models.

The estimates for all the available KO data and all suggested models obtained using the transition matrix in Eq. (2.27) are summarized as histograms in Fig. 2.19. Because of the different possibilities to combine the four different models in Eq. (2.23)-(2.26) and because of the different loci considered, in total there are 84 estimates for each KO data set. We plot the number of occurrences N of ψ_L (left) and ψ_R (right) in different ranges for both sorts of KO data (Dnmt1KO and Dnmt3a/b DKO).

The estimates of ψ_L spread over the whole interval $[0, 1]$ while in the case of ψ_R, nearly all estimates are larger than 0.99 and only in a few cases the dependence parameter is significantly smaller than 1. Hence, in most cases the methylation probabilities are independent of the right neighbor for both Dnmt1KO and Dnmt3a/b DKO. For ψ_L the dependence parameter in the Dnmt3a/b DKO case occurs more often close to 1, meaning that the transitions induced by Dnmt1 have little to no dependence on the left neighbor. On the other hand for Dnmt1KO the dependence parameter occurs more often at smaller values giving evidence that there is a dependence on the left neighbor for the activity of Dnmt3a/b. Note that all models show a similar behavior in terms of the dependence parameters for a given locus or position within a locus respectively, i.e. either $\psi_i \approx 1$ or $\psi_i < 1$ for all models. Since the histograms for Dnmt3a/b DKO

Table 2.4: Color scheme and fitting parameters (mean and standard deviation) for the Gaussian fits of the posteriors shown in Fig. 2.18.

		$n = 100$		$n = 1\,000$	
N_2	color	mean	sd	mean	sd
0	black	0.8052	0.1168	0.7743	0.1187
10^2	yellow	0.8059	0.1149	0.7888	0.1178
10^3	green	0.8193	0.0877	0.7841	0.1091
10^4	brown	0.8023	0.0311	0.8018	0.0442
10^5	cyan	0.8009	0.0163	0.8021	0.0185
10^6	black	0.8008	0.0098	0.7989	0.0102

look very similar for ψ_L and ψ_R, we used a two-sample Kolmogorov-Smirnov test to assess if they differ significantly. The resulting p-value of 1 indicates that there is no significant difference in this case. Note that we also get quite high p-values (0.786 and 0.433) when applying the test to the Dnmt1KO histogram for ψ_R and the two Dnmt3a/b DKO histograms. On the other hand, the p-values are significantly smaller for the Dnmt1KO ψ_L histogram, with a minimum of 0.019, indicating a different behavior for the dependence on the left neighbor for Dnmt3a/b.

Since ψ_R is usually close to 1 a smaller model with only three parameters $\theta = (\mu, \psi, \tau)$ can be proposed, where ψ is a dependence parameter for the left neighbor. This model can either be obtained by fixing $\psi_R = 1$ in the original model and setting $\psi = \psi_L$ or by redefining the transition probabilities to ψx if the left neighbor is unmethylated and $1 - \psi(1 - x)$ if the left neighbor is methylated. In that case ψ and ψ_L are related via $\psi = 0.5(\psi_L + 1)$. Note that both versions yield the same results. In order to check whether there is a significant difference in the original and the smaller model, we performed a Likelihood-ratio test with the null hypothesis that the smaller model is a special case of the original model. Since the original model with more parameters is always as least as good as the smaller model, our goal is to check in which cases the smaller model is sufficient. Indeed, if ψ_R was estimated to be approximately 1 the Likelihood-ratio test indicates that the smaller model is sufficient (p-value ≈ 1). On the other hand, for the few cases where ψ_R differs significantly from 1 the original model has to be used (p-value < 0.01).

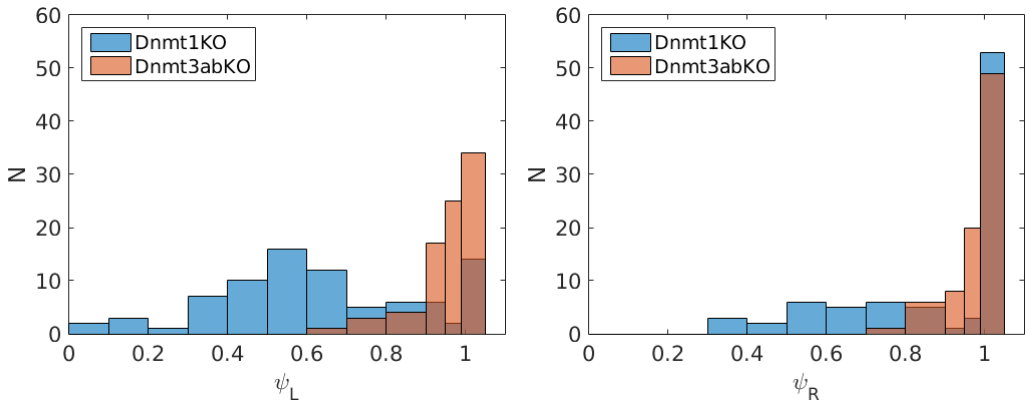


Figure 2.19: Histograms for the estimated dependence parameters ψ_L and ψ_R for all sets of three adjacent CpGs in all loci and for all suggested models.

Comparison of the Parameter Estimation Methods

As demonstrated for artificial data with known parameters, all methods can be used to successfully estimate the parameters given samples of sufficient sizes and show a similar performance in terms of accuracy. However, each of the methods comes with its own (dis)advantages.

While GMM and MLE have a clear optimization task (minimizing a score function for GMM and maximizing the likelihood for MLE) for the parameter estimation, in ABC there is more freedom in how to obtain the parameter estimations from the (approximated) posterior. Since ABC is a sampling based method and furthermore one may use summary statistics, which reduce the dimension of the needed information, in order to decide which parameters to accept into the posterior, it is especially appealing for systems with more CpGs, i.e. larger state spaces. The GMM estimator has a similar advantage since instead of the whole distribution, only some selected moments are needed for the parameter estimation. However, a finite set of moments does not necessarily capture all details in the behavior of the underlying distribution and can therefore only be considered to be an approximation.

In terms of runtime and memory requirements, ABC has low memory requirements since only the accepted parameters in the posterior have to be stored, but may suffer from long runtimes due to high rejection rates and the many samples needed. The typical number of function evaluations for ABC is in the order of 1 000-10 000, since the posterior has to be updated sufficiently often. For the optimization tasks for MLE and GMM the typical order of function evaluations is about 100. For GMM the moments can either be directly obtained from the model, if the respective analytic expressions are available or they could

be sampled otherwise. Sampling the likelihood is usually not preferable due to the small order of magnitude for the values in the distribution. It would require many samples to keep the statistical inaccuracies low enough, which would lead to very long runtimes. The numerical solution of the model to obtain the distribution and hence the likelihood is very fast, however the transition matrix has to be stored, which requires a lot of memory especially for a larger number of CpGs and therefore becomes infeasible for larger systems.

Since for the investigated scenarios the numerical solutions are available, all parameter estimations stem from a MLE if not stated otherwise in the remainder of this thesis.

2.5.2 CpG Distances

Since in general the distances between two adjacent CpGs are not identical for all pairs due to different DNA sequences, the distance may influence the neighborhood dependencies. We therefore take a closer look at the estimated dependence parameters shown in the histograms in Fig. 2.19 and link the parameters to their respective loci and distances between adjacent CpGs in base pairs (bps). The results for the estimation of the left and right dependence parameter for both Dnmt3a/b DKO and Dnmt1KO data, based on the transition matrix in Eq. (2.23) (model 1) are shown in Fig. 2.20. The results based on the other transition matrices yielded similar results and are therefore not presented here. The coloring of the symbols for the different loci is as follows: mSat (red), Afp (blue), IAP (green), L1 (pink) and Tex13 (black). As already seen before, in all cases, except for the dependence of the activity of Dnmt3a/b on the left neighbor, the dependence parameter is always close to 1, independent of the distance between the CpGs, i.e. the majority of the estimates for the dependence parameters fall into the interval $0.9 < \psi < 1$. Only Dnmt3a/b shows a stronger dependence on the left neighbor, i.e. in most cases $\psi < 0.9$, but no simple relation to the distance is visible. Another observation from Fig. 2.20 (c) is that the dependency parameters show very similar behaviors within the same locus. However, it is impossible to draw reliable conclusions due to the small sample size within each locus.

A reasonable assumption is that for (very) large distances between the CpGs the dependence on the neighbor state should eventually vanish. However, since the maximum distance is very small for our data (< 50 bps) we are not able to verify this assumption. Also note that an increasing distance in bps does not necessarily imply an increasing real distance, especially for small distances. Due to the double-helix geometry of the DNA, a CpG with a larger bps distance could be physically closer to another CpG than a CpG with a smaller bps dis-

tance. In the limit however, if the bps distance goes to infinity, so does the real distance. Note that we do not account for such effects in our model.

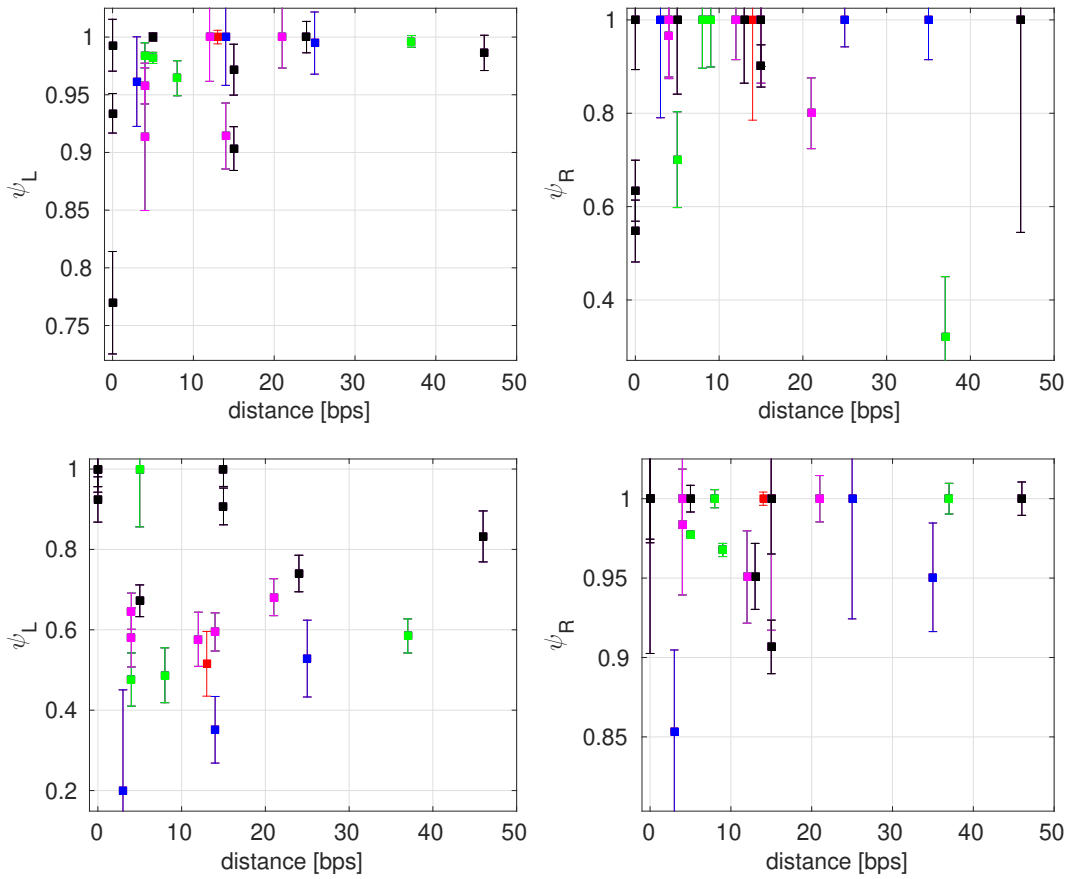


Figure 2.20: Dependence parameter versus distance between CpGs measured in base pairs (bps). The top row shows the results for the Dnmt3a/b DKO data, the bottom row for Dnmt1KO for model 1. The left (right) column shows results for the dependence parameter to the left (right). The different colors of the symbols represent the different loci and are explained in the main text. Note the different ranges on the Y axes.

2.5.3 Wild-Type Prediction

As a next step we use the estimated parameters from the KO data to predict the WT data. We have to employ the MLE twice in order to estimate the parameter vector $\hat{\theta}_1$ for Dnmt1 from the 3a/b DKO (double knockout) data and the vector $\hat{\theta}_{3a/b}$ for Dnmt3a/b from the Dnmt1KO data, where transition matrix (2.27) is used. The corresponding time instants are $t = 26$ for the 3a/b DKO data and $t = 41$ for the 1KO data. Instead of bootstrapping, we approximate the standard deviations of the estimated parameters $\hat{\theta}$ as follows: Let $\mathcal{I}(\hat{\theta}) = \mathbb{E}[-\mathcal{H}(\hat{\theta})]$ be the expected Fisher information, with the Hessian $\mathcal{H}(\hat{\theta}) = \nabla\nabla^\intercal \ell(\hat{\theta})$. The inverse of the expected Fisher information is a lower bound for the covariance matrix of the MLE such that we can use the approximation $\sigma(\hat{\theta}) \approx \sqrt{\text{diag}(-\mathcal{H}(\hat{\theta}))}$.

A prediction for the wild-type can be computed by combining the estimated vectors such that in the model both types of enzymes are active. For this, we insert $\hat{\theta}_1$ in P_s and $\hat{\theta}_{3a/b}$ in $\tilde{P}_s$ in (2.28) to obtain the transition matrix for the wild-type.

As a reminder, the models from Eq.(2.23)-(2.26) are referred to as *models 1-4*. For the prediction, the notation (x, y) is used to refer to Model x for the Dnmt3a/b DKO (only Dnmt1 active) and Model y for the Dnmt1KO case (only Dnmt3a/b active). One instance of the prediction, for which Model 1 was used for both Dnmt1KO and Dnmt3a/b DKO, i.e. $(1, 1)$, are shown in Fig. 2.21. Note that all wild-type predictions yielded a very similar accuracy. We list the corresponding estimations for the parameters for an example of a single copy gene (Afp) and a repetitive element (L1) in Tab. 2.5. Note that due to the fact that we only consider sequences of three CpGs here, the samples sizes are bigger than those for the whole sequence given in Tab. 2.2. While the standard deviation of the estimated parameters for μ is always of the order 10^{-2} and for τ of order 10^{-3}, it is usually of order 10^{-2} for ψ_i. Depending on the model, locus and position, standard deviations up to order 10^{-1} may occur for the dependence parameters in a few cases.

In Fig. 2.21 the predictions for the pattern distribution together with the WT pattern distribution and a prediction from the neighborhood independent model ($\psi_L = \psi_R = 1$) for all loci are shown in the main plot. As an inset the distributions are shown on a smaller scale to display small deviations. With the exception of patterns 1 and 64 (which corresponds to no methylation/full methylation of all sites) in L1 and pattern 64 in all loci, where the difference between WT and the numerical solution is about 10%, the difference is always small ($< 5\%$) as seen in the insets. In order to compare the performance of the neighborhood dependent and neighborhood independent model, we compute

the Kullback-Leibler divergence

$$KL = \sum_{j=1}^{4^L} \pi_j(\text{WT}) \log\left(\frac{\pi_j(\text{WT})}{\pi_j(\text{pred})}\right) \tag{2.67}$$

for both cases and each locus and list the results in Tab. 2.6. The mean and standard deviation were obtained via bootstrapping of the wild-type data ($10\,000$ bootstrap samples). The results show that the mean of KL as well as its standard deviation are always smaller for the neighborhood dependent model, i.e. the neighborhood dependent model yields more accurate predictions.

For the 16 proposed models from Eq. (2.28) we observe a similar performance for all loci and positions in terms of accuracy of the prediction. On the large scale the differences are not visible and even for the smaller scale the differences are small. We therefore only show two examples for mSat in Fig. 2.22, namely $(1,1)$ on the left and $(4,4)$ on the right. The distribution for all 16 possibilities can be found in Fig. A.1 in Appendix A.

By comparing KL that we list in Tab. 2.7, the similar performance of all 16 models can clearly be seen. The difference in KL between the "best" and the "worst" case is about 0.01. Again, the mean and standard deviation for KL were obtained via bootstrapping of the wild-type data ($10\,000$ bootstrap samples for each model). Since no confidence intervals of the parameters are included, this standard deviation can be regarded as a lower bound. However, even with these lower bounds the intervals of KL overlap for all models, such that no model can be favorized.

Table 2.5: Estimated parameters for the KO data and model $(1,1)$ based on Eq. (2.23) for the loci Afp and L1 with sample size N.

Afp

KO	μ	ψ_L	ψ_R	τ	N
1	0.452 ± 0.062	0.383 ± 0.076	1.000 ± 0.094	0.091 ± 0.016	134
3a/b	0.990 ± 0.003	0.984 ± 0.011	1.000 ± 0.006	$10^{-10} \pm 0.011$	186

L1

KO	μ	ψ_L	ψ_R	τ	N
1	0.334 ± 0.051	0.576 ± 0.067	1.000 ± 0.122	0.038 ± 0.004	1 047
3a/b	0.789 ± 0.037	1.000 ± 0.038	0.984 ± 0.045	$10^{-10} \pm 0.002$	805

Table 2.6: Kullback-Leibler divergence KL for the neighborhood dependent and independent predictions at all loci.

Locus	KL_{dep}	KL_{ind}
Afp	0.6820 ± 0.0914	3.3557 ± 0.0979
mSat	0.1398 ± 0.0134	0.2582 ± 0.0286
IAP	0.3615 ± 0.0482	0.5390 ± 0.0602
L1	0.5342 ± 0.0638	0.5639 ± 0.0771
Tex13	1.3364 ± 0.3235	2.0120 ± 0.3637

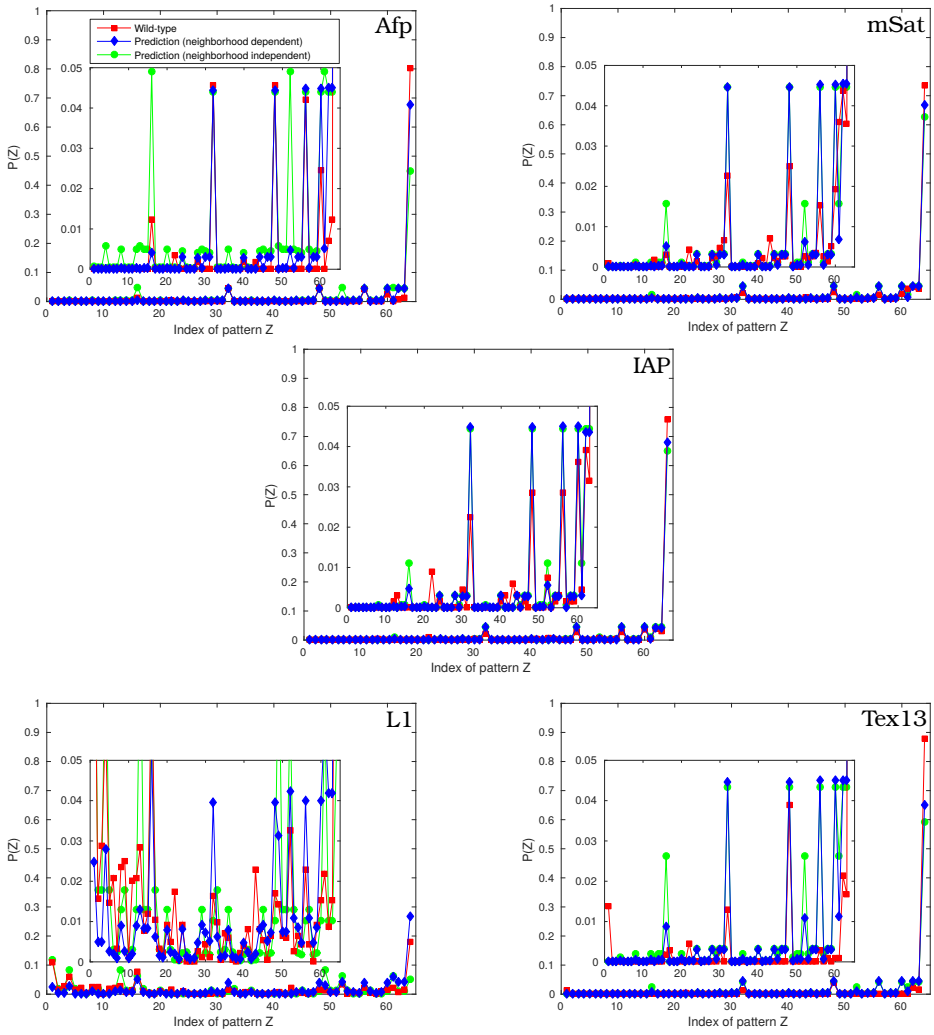


Figure 2.21: The figures show an example for the predicted (neighborhood dependent $(1, 1)$ and neighborhood independent) and the measured pattern distribution for each locus. The inset shows a zoomed in version of the distribution.

Table 2.7: Kullback-Leibler divergence KL for all 16 models at the locus mSat.

Model	KL	Model	KL
$(1,1)$	0.1398 ± 0.0134	$(3,1)$	0.1399 ± 0.0134
$(1,2)$	0.1398 ± 0.0134	$(3,2)$	0.1399 ± 0.0134
$(1,3)$	0.1398 ± 0.0134	$(3,3)$	0.1398 ± 0.0133
$(1,4)$	0.1337 ± 0.0127	$(3,4)$	0.1337 ± 0.0127
$(2,1)$	0.1438 ± 0.0137	$(4,1)$	0.1410 ± 0.0137
$(2,2)$	0.1439 ± 0.0136	$(4,2)$	0.1411 ± 0.0136
$(2,3)$	0.1439 ± 0.0137	$(4,3)$	0.1409 ± 0.0135
$(2,4)$	0.1374 ± 0.0133	$(4,4)$	0.1349 ± 0.0130

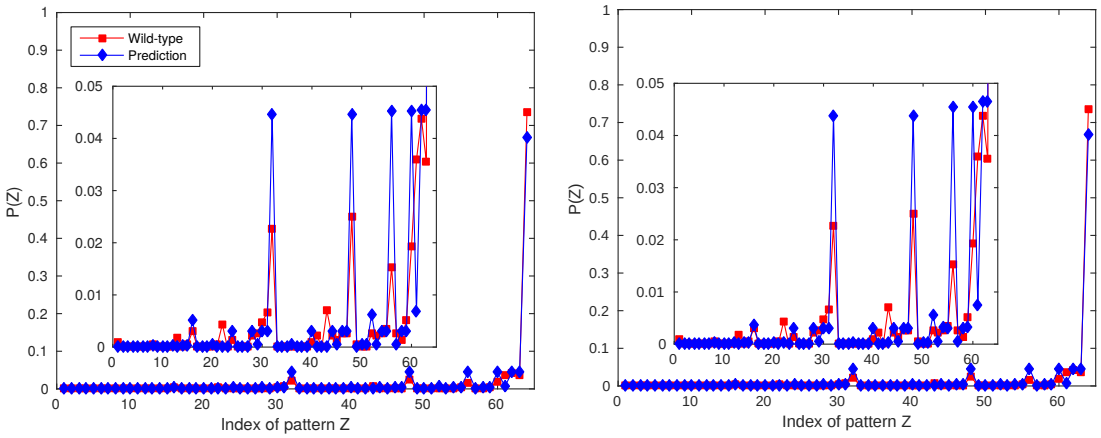


Figure 2.22: The figures show the predicted and the measured pattern distribution for two (left: $(1,1)$, right: $(4,4)$) of the 16 models for mSat. The inset shows a zoomed in version of the distribution. The red WT distribution is the same in both plots. Note the slight differences in both predictions for example in pattern 16, 62 and 63.

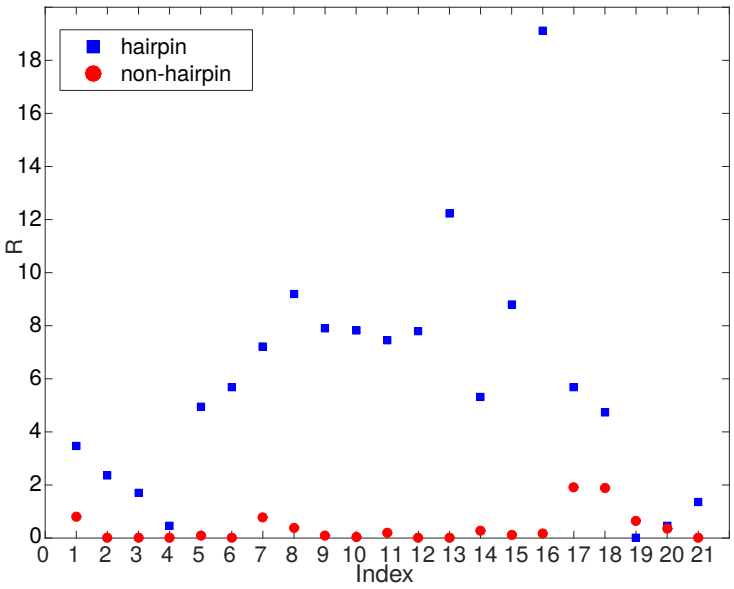


Figure 2.23: Ratio $R = \mu/\tau$ between maintenance and *de novo* rate for hairpin (blue) and non-hairpin data (red) for all loci. The loci are mapped to the indices as follows: mSat:1, Afp:2–4, IAP:5–8, L1:9–13, Tex13:14–21.

2.5.4 Non-Hairpin Data

So far we restricted the usage of the model to hairpin data, i.e. for one DNA molecule the methylation state of both strands is measured. For non-hairpin data there is only knowledge available for each strand independently. The information which strands stem from the same chromosome is not known. However, it is possible to compute the product of the likelihoods of the individual strand patterns, which resembles the likelihood of real hairpin data (assuming independence). Our results show that this approach works well as long as the states of the opposite strand do not determine the transition probabilities, which is the case for Dnmt1KO data, since Dnmt3a/b shows only little maintenance activity. Since Dnmt1's main activity is maintenance, we indeed found that the WT and Dnmt3a/b DKO data does not yield good results (results not shown).

To compare the performance of the model for hairpin and non-hairpin data, we split the original hairpin data in upper and lower strand and computed the product of likelihoods for the patterns using the independence assumption. We then estimated the parameters via MLE with our model and the computed distributions. We found that for Dnmt3a/b the results are very close to the original hairpin data in terms of dependence parameter ψ_L and ψ_R, since in the model definition these parameters rely only on information on the same strand. No information from the opposite strand influences the dependence

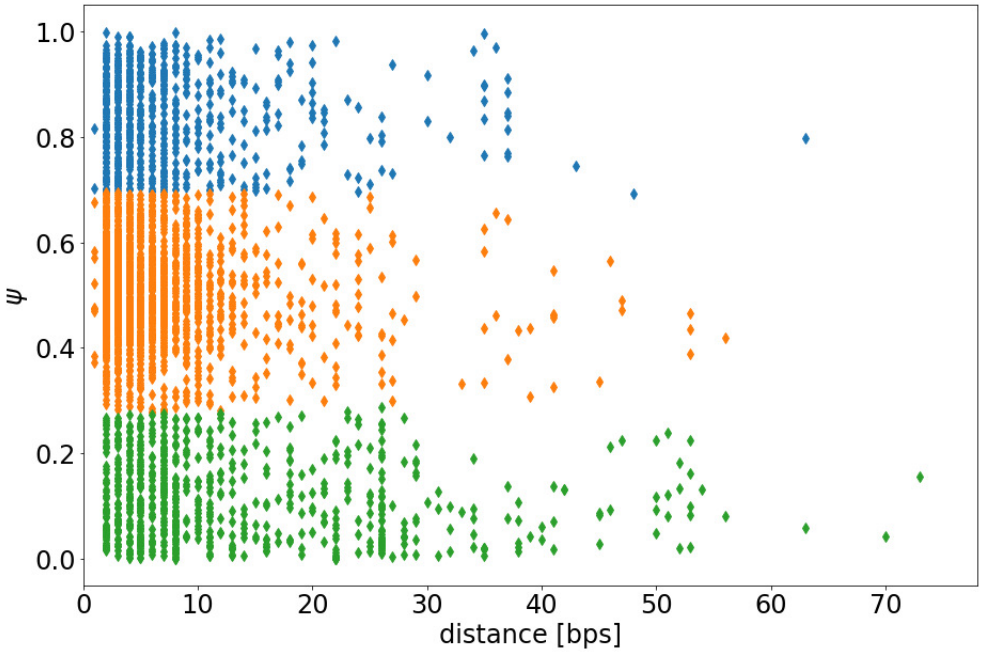


Figure 2.24: Dependence parameter versus distance between CpGs for the genome-wide data. The three colors represent three clusters. Cluster 0: blue, cluster 1: orange, cluster 2: green.

parameters. The ratio $R = \mu/\tau$ is usually smaller, i.e. the maintenance is underand the *de novo* activity overestimated, for the non-hairpin data as shown in Fig. 2.23. However, this does not lead to contradictory results since maintenance and *de novo* methylation can not be distinguished by the model if the CpG on the opposite strand is methylated.

2.5.5 Genome-Wide Data

Due to the limited amount of CpGs for the experiments in the previous sections, we also considered genome-wide hairpin data obtained from mouse embryonic stem cells to substantially increase the number of measured CpGs and hence also the number of possible distances between adjacent CpGs. In the genome-wide data the methylation state of the CpGs were recorded in windows of approximately 150 bps for a subset of CpGs, such that there is information available for about 4 million CpGs of the entire genome. The data contains the methylation state of each CpG and the position on the DNA, from which the distance between adjacent CpGs can be derived. For our analysis, we only consider CpGs within the sames read i.e. in the 150 bp window. This last in-

formation is of great importance since we want to investigate the neighborhood dependence and have to ensure that the three adjacent CpGs stem from the same DNA molecule. Therefore the data is filtered such that we omit all CpGs which do not form a sequence of at least three consecutive CpGs within one read. Note that we do not consider all cases where either only one or two CpGs were covered in the measurement window or because of missing CpGs the consecutive sequence is split in chunks of two CpGs or smaller. Furthermore we only considered CpG triples for which at least 64 (i.e. the number of possible patterns) measurements were taken. After applying these constrains there are $3\,489$ CpG triples left.

Since only WT data (and no KO data) was available for the whole genome, we had to use a modified version of the parameter estimation based on Eq. (2.28), which contains eight parameters (four for each enzyme). In order to reduce the model complexity we use the observations from the previous experiments, namely that only Dnmt3a/b shows a dependence to the left, and we therefore set the remaining dependence parameters $\psi_L^{(1)}$, $\psi_R^{(1)}$ and $\psi_R^{(3a/b)}$ to 1. The conversion errors for the data set are $c = 0.996$ and $d = 0.93$. The conversion rates are derived from short synthetic DNA fragments containing different cytosine forms at definite positions. These oligos become part of the hairpin bisulfite library and therefore undergo the same treatment as the stem cell DNA. Thus, after sequencing, we can determine the conversion rate of C and 5mC independently of our biological sample.

Despite considering only CpG triples with a coverage of at least 64, in general the coverage is pretty low compared to the hairpin data used for the parameter estimation in the previous section. We therefore employ Bayesian inference rather than MLE for the parameter estimation in the genome-wide data [93]. We use a Metropolis Hastings algorithm with the estimations from ML as starting points and a Gaussian proposal distribution with mean 0 and a standard deviation of 0.01 such that on average 40% of the $5\,000$ total trials per CpG triplet are accepted for the posterior distribution. Afterwards a variant of the k-means algorithm is applied, which also considers standard deviations of the quantities that should be clustered [72]. Note that in order to avoid a domination by the much larger distances in the clustering, the distance is normalized before the algorithm is applied. The ideal number of clusters is chosen by minimizing the Davies-Bouldin index [24], which is defined as the ratio between cluster separation and similarity within the clusters. The results of the parameter estimation and the clustering is shown in Fig. 2.24.

Note that the clustering is based on dependence parameter and distance only. The methylation state is not an input of the clustering algorithm. Interestingly, the different clusters show different behavior in terms of methylation

levels, as well as in genomic location. CpGs from clusters 0 and 1 (no/low dependence, blue and orange) tend to be unmethylated and the majority is located in promoter regions, while CpGs from cluster 2 (high dependence, green) are usually fully methylated and located in intergenic regions (results not shown) [37, 86].

In our results the methylation state of a CpG shows a strong dependence on the methylation state of the left neighbor even for distances up to 70 bps. We therefore conclude that the independence starts at much larger distances. Note that due to the restriction that the three CpGs have to be within the same 150 bps window during the measurement, even for the genome-wide data the distances between the CpGs are rather short. It is therefore not possible with the current data and measurement techniques to check hypotheses such as the independence of neighboring CpGs for large distances. Furthermore, it is possible that the filtering introduces a bias in the available data in terms of genomic locations, since only regions are kept, where the sufficiently many CpGs are close together. Regions with CpGs further apart are usually sorted out and therefore underrepresented in our final data set. With more advanced measurement techniques in the future it should be possible to investigate adjacent CpGs with larger distances, which will also decrease the bias in the available locations.

2.6 Conclusion

2.6.1 Discussion

We proposed a set of stochastic models for the formation and modification of methylation patterns over time. These models take into account the state of the CpG sites in the spatial neighborhood and allow to describe different hypotheses about the underlying mechanisms of methyltransferases adding methyl groups at CpG sites. We introduced a stochastic automata networks description of our model, that allows to easily generate the transition probability matrix and also allows to generalize the model to different scenarios, like more modifications of cytosine other than 5mC or also different hypotheses about the working mechanisms of the Dnmts.

We used knockout data from bisulfite sequencing at several loci to learn the efficiencies at which these enzymes perform methylation. To estimate the efficiencies, we sucessfully tried different parameter estimation methods, namely the generalized method of moments, approximate Bayesian computation and maximum likelihood. Each of these methods comes with their own advantages and disadvantages, but all are feasible in certain situations.

By combining the efficiencies estimated with a maximum likelihood estimator, we accurately predicted the probability distribution of the patterns in the wild-type. Moreover, we found that in all cases the models predict values for the dependence parameters ψ_L and ψ_R close to 1 and therefore independence of methylation for the Dnmt3a/b DKO meaning that Dnmt1 methylates CpGs independent of the methylation of neighboring CpGs. For Dnmt3a/b on the other hand we could identify dependences on the neighboring CpGs. Both findings are in accordance with current existing mechanistic models: Dnmt1 reliably copies the methylation from the template strand to maintain the distinct methylation patterns, whereas Dnmt3a/b try to establish and keep a certain amount of CpG methylation at a given locus. Interestingly, our models only suggest dependences of *de novo* methylation activity on the CpGs in the 5′ neighborhood. This indicates that Dnmt3a and Dnmt3b show a preference to methylate CpGs in a 5′ to 3′ direction and could point towards a processive or cooperative behavior of these enzymes like recently described in *in vitro* experiments [28, 61].

Our results indicate that, at least for small distances, rather the genetic region than the distance determines the dependence on the neighbors. Compared to a neighborhood independent model with $\psi_L = \psi_R = 1$, a neighborhood dependent model shows better predictions and furthermore allows to investigate (possible) connections of adjacent CpGs and their methylation states. As long as no information from the opposite strand is needed, i.e., if the maintenance activity is not too high, as in the Dnmt1KO data, our model can also be used for non-hairpin data. Applying our model at genome-wide data reveals distinct dependence clusters with individual methylation patterns.

2.6.2 Future Work

So far, we considered a quite simple model, i.e., there are only four parameters when modelling one of the enzymes. However, there are some obvious extensions or generalizations. For example, there is only one *de novo* probability for both parental and daughter strand. There are existing models with separate probabilities. In our model this additional rate can also be easily included.

Another point is the transitions at CpGs at the left or right boundary. Here, we used a combination of the bulk probabilities, which are weighted by the average methylation level. Other possibilities are extra probabilities for the boundary cases, or it is also imaginable to infer the methylation state of the CpG left (right) of the left (right) boundary with suitable methods from machine learning and then use the bulk probabilities.

So far, the transition probabilities for methylation events depends only on the methylation states of the two adjacent CpGs on the same strand. But it is also

possible, that for example the diagonal neighbors, i.e., the methylation status of the adjacent CpGs on the opposite strand, or the methylation status of CpGs further away may have an influence on the transition probabilities. Also considering the diagonal neighbors or two CpGs to each side (on the same strand) would increase the number of possible neighborhood combinations from four to 16 and would also require additional dependence parameters. While for the diagonal neighbor case sequences of three CpGs are still sufficient, for the other case we would need sequences of length five, which would also increase the number of possible patterns to 1024. For this case it is oftentimes hard to get data with sufficiently deep coverage. Instead of introducing extra dependence parameters for each CpG in longer sequences, it may also be possible to find a dependence function, which also depends on the distance between adjacent CpGs to account for the (in general) different number of base pairs between two CpGs. A distance function would have the advantage, that the number of (dependence) parameters remains fixed and does not increase with a larger number of CpGs. Extra parameters or a distance function are also needed, when investigating multiple CpGs simultaneously, even when only considering the adjacent CpGs for the transition probabilities, since assuming the same dependencies for all CpGs is a strong assumption. It is very likely that each CpG has its own dependencies because of different distances to the next CpGs or due to different base sequences in the DNA. So far, all parameters are constant over time. Another possible generalization of the model is to introduce time-dependent efficiencies.

Also, there are more possible working mechanisms for the Dnmts, other than the processesive and distributive behavior that we described here. Here, in distributive methylation the attachment to the DNA may happen randomly at every site of the considered DNA sequence. In a similar approach, which is still characterized by frequent attachment and detachment from the DNA, the Dnmt performs a diffusive motion while unbound and is therefore more likely to reattach in the vicinity of the detachment site, rather than far away. We call this behavior *diffusive methylation*. There are also combinations of the described behaviors imaginable. For example, a Dnmt may perform processive methylation while attached to the DNA and show a diffusive behavior while unbound. A description like that would require additional states, i.e., whether a Dnmt is bound to the DNA or unbound. In that case it would also be possible to distinguish if a methylation event failed because the Dnmt was not attached to the DNA or because there is a imperfect methylation probability. So far when we estimate a methylation probability smaller than one, it is a combination of both effects, since our model is not able to distinguish both cases. However, it is chal-

lenging (if not impossible) to obtain biological data on the binding state of the Dnmts.

Finally, as already hinted in Section 2.3.5 about the stochastic automata network description, it is straightforward to generalize the model to more states per CpG, i.e., including further modifications of cytosine, like 5hmC, 5fC or 5caC. Furthermore we can easily switch to a hybrid description and omit the necessity of specifying the order of certain events beforehand.

Chapter 3

The Generalized Method of Moments for Chemical Reaction Networks

3.1 Introduction

A widely-used approach in systems biology research is to design quantitative models of biological processes and refine them based on both computer simulations and wet-lab experiments. While a large amount of sophisticated parameter inference methods have been proposed for deterministic models, only a few approaches allow the efficient calibration of parameters for large discrete-state stochastic models that describe stochastic interactions between molecules within a single cell. Since research progress in experimental measurement techniques that deliver single-cell and single-molecule data has advanced, the ability to calibrate such models is of key importance. For instance, the widely-used flow cytometric analysis delivers data from thousands of cells which yields sample means and sample variances of molecular populations.

Here, we focus on the most common scenario: a discrete stochastic model of a cellular reaction network with unknown reaction rate constants and population snapshot data such as sample moments of a large number of observed samples. The state of the model corresponds to the vector of current molecular counts, i.e., the number of molecules of each chemical species, and chemical reactions trigger state transitions by changing the molecular populations. A system of ordinary differential equations, the chemical master equation [94], describes the evolution of the state probabilities over time.

A classical maximum likelihood (ML) approach, in which the likelihood is directly approximated, is possible if all populations are small [3] or if the model

shows simple dynamics (e.g. multi-dimensional normal distribution with time-dependent mean and covariance matrix) such that the likelihood can be approximated by a normal distribution [97]. In this case, the likelihood (and its derivatives) can usually be approximated efficiently and global optimization techniques are employed to find parameters that maximize the likelihood. However, if large populations are present in the system then direct approximations of the likelihood are unfeasible since the underlying system of differential equations contains one equation for each state and the main part of the probability mass of the model distributes on an intractably large number of states. Similarly, if the system shows complex dynamics such as multimodality, approximations of the likelihood based on Gaussian distributions become inaccurate.

In the last years several methods have been developed to accurately simulate the moments of the underlying probability distribution up to a certain order m over time [2, 29, 115]. The complexity of these simulation methods is therefore independent of the population sizes but, for large m, the corresponding differential equations may become stiff and lead to poor approximations. However, reconstructions of complex distributions from their moments show that for many systems already for small m (e.g. $m \in \{4, \ldots, 8\}$) the moments contain sufficient information about the distribution such as the strength and location of regions of attraction (i.e. regions of the state space containing a large proportion of the probability mass) [4].

For models with complex distributions such as multiple modes or oscillations, the accuracy and the running time of the moment approximation can be markedly improved, when conditional moments are considered in combination with the probabilities of appropriately chosen system modes such as the activity state of the genes in a gene regulatory network [56, 57, 63, 95]. Recently a full derivation of the conditional moment equations was derived and numerical results show that when the maximum order of the considered moments is high, the number of equations that have to be integrated is usually much smaller for the conditional moments approach and the resulting equations are less stiff [55]. In addition, the approximated (unconditional) moments are more accurate when the same maximal order is considered.

An obvious parameter inference approach is the matching of the observed sample moments with those of the moment-based simulation of the model. Defining the differences between sample and (approximated) population moments as cost functions that depend on the parameters, an approach that minimizes the sum of the squared cost functions seems reasonable. However, in a simple least-squares approach low moments such as means and (co-)variances contribute equally to the sum of squared differences as higher moments, whose

absolute magnitudes are much higher (even if they are centralized). Moreover, correlations between the different cost functions may exist and thus necessitate an approach where also products of two different cost functions are considered.

The generalized method of moments (GMM) that is widely used in econometrics provides an estimator that is computed after assigning appropriate weights to the different cost function products [52]. The GMM estimator has, similar to the ML estimator, desirable statistical properties such as being consistent and asymptotically normally distributed. Moreover, for optimally chosen weights it is an asymptotically efficient estimator, which implies that (asymptotically) it has minimum variance among all estimators for the unknown parameters.

In this chapter, we explore the usefulness of the GMM for moment-based simulations of stochastic reaction networks. We focus on two particular estimators that are commonly used in econometrics: the two-step estimator of Hansen [52] and the demean estimator [50]. We study the accuracy and variance of the estimator for different maximal moment orders and different sample sizes by applying the GMM to two case studies. In addition, we show that poor approximations of some higher order moments have a strong influence on the quality of the estimation. Interestingly, we see that the additional information about the covariances of the cost functions can lead to identification of all parameters. Additionally, the variance of the estimator becomes smaller when higher order moments are included. Compared to the simple least-squares approach, the GMM approach yields very accurate estimates.

This chapter is organized as follows: In Section 3.2 we give the necessary theoretical background. The results are presented in Section 3.3. We discuss our findings with regards to related work in Section 3.4 and conclude this chapter in Section 3.5.

3.2 Methods

3.2.1 Stochastic Chemical Kinetics

Our inference approach relies on a Markov modelling approach that follows Gillespie's theory of stochastic chemical kinetics. We consider a well-stirred mixture of n molecular species in a volume with fixed size and fixed temperature and represent it as a discrete-state Markov process $\{\mathbf{X}(t), t \geq 0\}$ in continuous-time [42]. The random vector $\mathbf{X}(t) = (X_1(t), \ldots, X_n(t))$ describes the chemical populations at time t, i.e., $X_i(t)$ is the number of molecules of type $i \in \{1, \ldots, n\}$ at time t. Thus, the state space of $\mathbf{X}$ is $\mathbb{Z}_+^n = \{0, 1, \ldots\}^n$. The state changes of $\mathbf{X}$ are triggered by the occurrences of chemical reactions.

Each of the R different reaction types has an associated non-zero change vector $\mathbf{v}_j \in \mathbb{Z}^n$ ($j \in \{1, \ldots, R\}$), where $\mathbf{v}_j = \mathbf{v}_j^- + \mathbf{v}_j^+$ such that $\mathbf{v}_j^-$ ($\mathbf{v}_j^+$) contains only non-positive (non-negative) entries and specifies how many molecules of each species are consumed (produced) if an instance of the reaction occurs, respectively. Thus, if $\mathbf{X}(t) = \mathbf{x}$ for some $\mathbf{x} \in \mathbb{Z}_+^n$ with $\mathbf{x} + \mathbf{v}_j^-$ being non-negative, then $\mathbf{X}(t + dt) = \mathbf{x} + \mathbf{v}_j$ is the state of the system after the occurrence of the j-th reaction within the infinitesimal time interval $[t, t + dt)$. W.l.o.g. we assume here that all vectors $\mathbf{v}_j$ are distinct.

We use $\alpha_1, \ldots, \alpha_R$ to denote the propensity functions of the reactions, where $\alpha_j(\mathbf{x}) \cdot dt$ is the probability that, given $\mathbf{X}_t = \mathbf{x}$, one instance of the j-th reaction occurs within $[t, t + dt)$. Assuming law of mass action kinetics, $\alpha_j(\mathbf{x})$ is chosen proportional to the number of distinct reactant combinations in state $\mathbf{x}$. An example is given in Table 3.1, where the first reaction gives as change vectors, for instance, $\mathbf{v}_1^- = (-1, 0, 0)$, $\mathbf{v}_1^+ = (0, 1, 0)$, $\mathbf{v}_1 = (-1, 1, 0)$. Note that, given the initial state $\mathbf{x} = (1, 0, 0)$, at any time either the DNA is active or not, i.e. $x_1 = 0$ and $x_2 = 1$, or $x_1 = 1$ and $x_2 = 0$. Moreover, the state space of the model is infinite in the third dimension. Although our inference approach can be used for any model parameter in the sequel we simply assume that the proportionality constants c_j are unknown and have to be estimated based on experimental data.

For $\mathbf{x} \in \mathbb{Z}_+^n$ and $t \geq 0$, let $p_t(\mathbf{x})$ denote the probability $P(\mathbf{X}(t) = \mathbf{x})$. Assuming fixed initial conditions p_0 the evolution of $p_t(\mathbf{x})$ is given by the chemical master equation (CME) [94]

$$\frac{\partial}{\partial t} p_t(\mathbf{x}) = \sum_{j : \mathbf{x} - \mathbf{v}_j^- \geq 0} \alpha_j(\mathbf{x} - \mathbf{v}_j) p_t(\mathbf{x} - \mathbf{v}_j) - \alpha_j(\mathbf{x}) p_t(\mathbf{x}),$$

which is an ordinary first-order differential equation that has a unique solution under certain mild regularity conditions. Since for realistic systems the number of states is very large or even infinite, applying standard numerical solution techniques to the CME is infeasible. If the populations of all species remain small (at most a few hundreds) then the CME can be efficiently approximated using projection methods [58, 100] or fast uniformization methods [92, 114]. Otherwise, i.e., if the system contains large populations, then analysis methods with running times independent of the population sizes have to be used such as moment closure approaches [2, 29, 115] or methods based on van Kampen's system size expansion [65, 120]. For both approaches, accurate reconstructions of the underlying probability distribution, i.e., the solution of the CME, are possible [4, 120].

3.2.2 Moment-Based Analysis

From the CME it is straightforward to derive the following equation for the derivative of the mean of a polynomial function $T : \mathbb{Z}_+^n \to \mathbb{R}$ on $\mathbf{X}(t)$.

$$\begin{aligned} &\tfrac{d}{dt}E[T(\mathbf{X}(t))] \\ &= \sum_{j=1}^{R} E\left[\alpha_j(\mathbf{X}(t)) \cdot (T(\mathbf{X}(t) + v_j) - T(\mathbf{X}(t)))\right] \end{aligned} \tag{3.1}$$

Omitting the argument t of $\mathbf{X}$ and choosing $T(\mathbf{X}) = X_i, X_i^2, \ldots$ yields the following equations for the (exact) time evolution of the m-th moment $E[X_i^m]$ of the distribution for the i-th species.

$$\begin{aligned} &\tfrac{d}{dt}E[(X_i)^m] \\ &= \sum_{j=1}^{R} E[\alpha_j(\mathbf{X}) \cdot ((X_i + v_{ji})^m - (X_i)^m)], \end{aligned} \tag{3.2}$$

where v_{ji} refers to the i-th component of the change vector $\mathbf{v}_j$. In a similar way, equations for mixed moments are derived.

If all reactions are at most monomolecular ($1 \geq \sum_i |v_{ji}^-|$ for all j), then no moments of order higher than k appear on the right side (also in the mixed case) and we can directly integrate all equations for moments of at most order m. However, most systems *do* contain bimolecular reactions (in particular those with complex behavior such as multistability). In this case we consider a Taylor expansion of the multivariate function

$$f(\mathbf{X}) = \alpha_j(\mathbf{X}) \cdot (T(\mathbf{X} + v_j) - T(\mathbf{X}))$$

about the mean $\mu := E[\mathbf{X}]$. It is easy to verify that, when applying the expectation to the Taylor sum, the right side only contains derivatives of f at $\mathbf{X} = \mu$, which are multiplied by central moments of increasing order. For instance, for $m = 1$ and a single species system with $n = 1$, Eq. (3.2) becomes

$$\begin{aligned} \tfrac{d}{dt}E[(X_i)] &= \sum_{j=1}^{R} v_{ji} E[\alpha_j(\mathbf{X})] \\ &= \sum_{j=1}^{R} v_{ji}\big(\alpha_j(\mu) + \tfrac{E[(\mathbf{X}-\mu)]}{1!} \cdot \tfrac{\partial}{\partial x}\alpha_j(\mu) \\ &\qquad + \tfrac{E[(\mathbf{X}-\mu)^2]}{2!} \cdot \tfrac{\partial^2}{\partial x^2}\alpha_j(\mu) + \ldots\big) \end{aligned}$$

In the expansion, central moments of higher order may occur. For instance, in the case of bimolecular reactions, the equations for order m moments involve

central moments of order $m+1$ since second order derivatives are non-zero. By converting the non-central moments to central ones and truncating the expansion at some fixed maximal order m, we can close the system of equations when we assume that higher order central moments are zero. A full derivation of the moment equations using multi-index notation (as required for $n > 1$) can be found in [29].

The accuracy of the inference approach that we propose in the sequel depends not only on the information given by the experimental data but also on the accuracy of the approximated moments. Different closure strategies have been suggested and compared in the last years showing that the accuracy can be improved by making assumptions about the underlying distribution (e.g. approximate log-normality) [13, 113]. In addition, the accuracy of moment-closure approximations has been theoretically investigated [46].

3.2.3 Hybrid Approaches

Compared to deterministic models that describe only average behaviors, stochastic models provide interesting additional information about the behavior of a system. Although this comes with additional computational costs, it is in particular for systems with complex behavior, such as multimodality or oscillations, of great importance. Often the underlying sources of multiple modes are discrete changes of gene activation states that are described by chemical species whose maximal count is very small (e.g. 1 for the case that the gene is either active, state 1, or inactive, state 0). Then the moment-based approaches described above can be improved (both in terms of accuracy and computation time) by considering conditional moments instead [55, 56, 57, 95, 116]. The idea is to split the set of species into species with small and large populations and consider the moments of the large populations conditioned on the current count of the small populations. For the small populations, a small master equation has to be solved additionally to the moment equations to determine the corresponding discrete distribution. More specifically, if $\hat{\mathbf{x}}$ is the subvector of $\mathbf{x}$ that describes the small populations and $\tilde{\mathbf{x}}$ is the subvector of the large populations (i.e. $\mathbf{x} = (\hat{\mathbf{x}}, \tilde{\mathbf{x}})$), then for the distribution of $\hat{\mathbf{x}}$ we have

$$\begin{aligned}\frac{d}{dt}p_t(\hat{\mathbf{x}}) = &\sum_{j:\hat{\mathbf{x}}-\hat{\mathbf{v}}_j\geq 0} E[\alpha_j(\mathbf{X}) \mid \hat{\mathbf{X}} = \hat{\mathbf{x}} - \hat{\mathbf{v}}_j]p_t(\hat{\mathbf{x}} - \hat{\mathbf{v}}_j) \\ &- \textstyle\sum_j E[\alpha_j(\mathbf{X}) \mid \hat{\mathbf{X}} = \hat{\mathbf{x}}]p_t(\hat{\mathbf{x}})\end{aligned}$$

where $\hat{\mathbf{v}}_j$ is the corresponding subvector of $\mathbf{v}_j$. Using Taylor expansion, the conditional expectations of the propensities can, as above, be expressed in terms of conditional moments of the large populations. In addition, equations for the

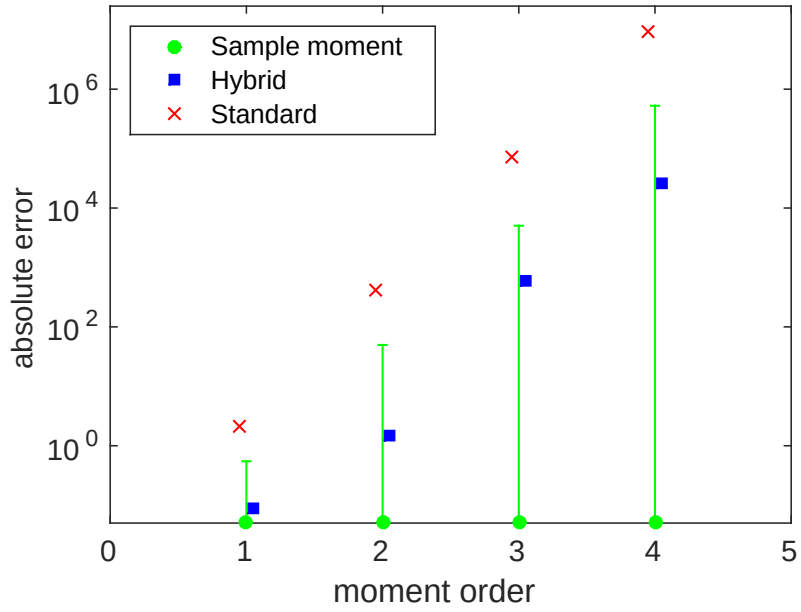


Figure 3.1: Absolute error of the first four moments of P_1 for the exclusive switch model, where the moments are either computed based on a standard moment closure approach or a hybrid approach. The maximal order of the considered moments is 5.

conditional moments of the large populations can be derived in a similar way as above. For instance, the partial mean $E[\tilde{X}_i \mid \hat{x}]p_t(\hat{x})$ follows the time evolution

$$\begin{aligned}
&\tfrac{\partial}{\partial t}\left(E[\tilde{X}_i \mid \hat{x}]p_t(\hat{x})\right) \\
&= \sum_{j:\hat{\mathbf{x}}-\hat{\mathbf{v}}_j\geq 0} E[(\tilde{X}_i + v_{ij})\alpha_j(\mathbf{X}) \mid \hat{\mathbf{X}} = \hat{\mathbf{x}} - \hat{\mathbf{v}}_j]p_t(\hat{\mathbf{x}} - \hat{\mathbf{v}}_j) \\
&- \textstyle\sum_j E[\tilde{X}_i\alpha_j(\mathbf{X}) \mid \hat{\mathbf{X}} = \hat{\mathbf{x}}]p_t(\hat{\mathbf{x}})
\end{aligned}$$

where on the right side again Taylor expansion can be used to replace unknown conditional expectations by conditional moments. As above a dependence on higher conditional moments may arise and a closure approach has to be applied to arrive at a finite system of equations. Unconditional moments can then be derived by summing up the weighted conditional moments. It is important to note that if $p_t(\hat{x}) = 0$ then algebraic equations arise turning the equation system into a system of differential-algebraic equations, which renders its solution more difficult (see [55, 68] for details).

In Fig. 3.1 we give an example for a comparison of the accuracy of the hybrid approach and the standard moment closure (assuming that all central moments above a fixed maximal order are zero) for one of our case studies. As "exact" moment values we chose the average of 500 000 samples generated by the stochastic simulation algorithm (SSA) [41] and considered the absolute difference to the approximated moments of one chemical population until a maximal order of four. Since for our case studies we assumed 10 000 samples we additionally plot the (approximated) standard deviation of the 50 sample means taken from batches of 10 000 samples. The moments computed based on the hybrid approach show a smaller error than those computed using the standard moment closure and lie within the deviations given by the sample moments. For the example in Fig. 3.1 we have 126 equations for the standard approach up to an order of four. In the hybrid case there are 14 moment equations and one equation for the mode probability per mode leading to a total number of 45 equations. However, reductions are possible for the standard approach when the model structure is exploited [111]. We do not make use of these reductions here but choose the hybrid approach mainly because it gives more accurate results for the (unconditional) moments. This strongly improves the quality of the estimated parameters as demonstrated in Section 3.3.

3.2.4 The Generalized Method of Moments Revisited

We assume that observations of a biochemical network were made using single-cell analysis that gives population snapshot data (e.g. from flow cytometry measurements). Typically, large numbers (about 5 000-10 000 [51, 54, 99]) of independent samples can be obtained where every sample corresponds to one cell. It is possible to simultaneously observe one or several chemical populations at a time in each single cell. In the sequel, we first describe the inference procedure based on the protein counts for a single observation time point and a single chemical species that is observed. Later, we extend this to several time points and species.

Note that, in contrast to Section 2.4.1 where we had to define special moments, we can use ordinary moments here. Since the r-th entry in the vector $\mathbf{Y}$ contains the r-th power of the protein count here, we remove the parentheses in the exponent. For a fixed measurement time t we can define the r-th order sample moment as

$$\overline{Y^r} = \frac{1}{N} \sum_{k=1}^{N} Y_k^r, \tag{3.3}$$

where Y_k is the k-th sample of the observed molecular count and there are N samples in total. For large N, the sample moments are asymptotically unbiased estimators of the population moments.

Let θ be a vector of, say, $q \leq m$ unknown reaction rate constants[1], for which some biologically relevant range is known. Moreover, let m_r be the r-th theoretical moment, i.e., $m_r(\theta) := E[Y_k^r]$. In the sequel we also simply write Y instead of Y_k whenever Y appears inside the expectation operator or when the specific index of the sample is not relevant.

Besides the (possibly) large variance of the least squares estimator defined in Eq. (2.50) and the possible correlations between the cost functions $g_r(\theta) = \overline{Y^r} - m_r(\theta)$, we face an additional problem here: In contrast to Section 2.4.1, where all moments had (roughly) the same order of magnitude, the higher-order moments of the protein counts are much larger than the low-order moments. Therefore, without suitable weights the higher-order moments would dominate the cost functions and hence strongly affect the estimation process. Additionaly, for increasing moment order the variance of the sample moments increases and so does the variance of the estimator. Furthermore, since we use only approximations for the theoretical moments here, we introduce an addiational source of possible errors, which may influence the accuracy of the estimator. Therefore, the weights in the score function are of utmost importance.

As a reminder, the GMM estimator is defined as

$$\hat{\theta} = \arg\min_{\theta} \mathbf{g}(\theta)' W \mathbf{g}(\theta),$$

with the weight matrix $W \propto F^{-1}$, where $F(\theta_0) = COV[\mathbf{Y}, \mathbf{Y}]$ will give the estimator with the smallest variance. We also discussed the conditions for identifiability, consistency properties and the demean estimator, which subtracts the sample means instead of the theoretical moments in the sample counterpart of the covariance in Section 2.4.1.

Another possibility to obtain an ideal weight matrix is the so-called *multi-step approach*, which is also called *iterated GMM estimator* [53]). Since F depends on the (unknown) true value θ_0, W is chosen as the identity matrix I in the first step and an initial estimate $\tilde{\theta}_1$ is computed. In the later steps, F is estimated by the sample counterpart of $E[\mathbf{f}(Y, \tilde{\theta})\mathbf{f}(Y, \tilde{\theta})^T]$, i.e.,

$$\hat{F}_1(\tilde{\theta}_{\ell-1}) = \frac{1}{N} \sum_{k=1}^{N} \mathbf{f}(Y_k, \tilde{\theta}_{\ell-1})\mathbf{f}(Y_k, \tilde{\theta}_{\ell-1})^T, \tag{3.4}$$

where $\tilde{\theta}_{\ell-1}$ is the estimate from the previous step. In this way, the estimation of the parameter and the weight matrix is gradually improved. Note that for each

[1]It is straightforward to adapt the approach that we present in the sequel to the case that other unknown continuous parameters have to be estimated.

step, we have to run a separate optimization. Usually a two-step approach is already sufficient.

In demean and multi-step the weight matrix can be computed beforehand and is fixed during the optimization itself. A third way to define an estimator is to include a parameter-dependent (not fixed) weight matrix into the optimization, i.e.,

$$\hat{\theta} = \arg\min_{\theta} \mathbf{g}(\theta)' W(\theta) \mathbf{g}(\theta), \tag{3.5}$$

with the weight matrix $W(\theta) = (\hat{F}_1(\theta))^{-1}$, where $\hat{F}_1$ is defined in Eq. (3.4). In this case, the weight matrix has to be recomputed in each optimization step. This estimator is called the *continuously updating GMM estimator* [53].

The estimation procedure described above can be generalized to several dimensions by also using mixed sample moments instead of only $\overline{Y^r}$ and mixed theoretical moments instead of only $m_r(\theta)$. For instance, for moments up to order two and two simultaneously observed species X and Y, we use the cost functions

$$\begin{aligned}
g_1(\theta) &= \tfrac{1}{N}\textstyle\sum_{k=1}^{N} X_k - E[X \mid \theta] \\
g_2(\theta) &= \tfrac{1}{N}\textstyle\sum_{k=1}^{N} Y_k - E[Y \mid \theta] \\
g_3(\theta) &= \tfrac{1}{N}\textstyle\sum_{k=1}^{N} X_k Y_k - E[XY \mid \theta] \\
g_4(\theta) &= \tfrac{1}{N}\textstyle\sum_{k=1}^{N} X_k^2 - E[X^2 \mid \theta] \\
g_5(\theta) &= \tfrac{1}{N}\textstyle\sum_{k=1}^{N} Y_k^2 - E[Y^2 \mid \theta].
\end{aligned}$$

In the same way, we can extend the estimators $\hat{F}_1$ and $\hat{F}_2$ to several dimensions. For instance, the covariance between $X_k Y_k$ and X_k^2 can be estimated as

$$\frac{1}{N}\sum_{k=1}^{N}(X_k Y_k - \overline{XY})(X_k^2 - \overline{X^2}),$$

where again we use $\overline{*}$ to denote the sample mean operator.

If, instead of snapshot data for a single observation time, independent samples for different times are available then the GMM estimator can also be easily generalized to

$$\hat{\theta} = \arg\min_{\theta} \sum_{t=t_0}^{t_f} \mathbf{g}^{(t)}(\theta)' W^{(t)} \mathbf{g}^{(t)}(\theta). \tag{3.6}$$

Here, for each time point $t \in \{t_0, \ldots, t_f\}$ the vector of cost functions $\mathbf{g}^{(t)}$ is calculated as before and the minimum is taken over the sum of these uncorrelated cost functions. Note that for each observation time point a weight matrix $W^{(t)}$ has to be computed. In the two-step approach, the initial weight matrices are

all equal to the identity matrix and then in the second step different weight matrices may arise since the estimator of F depends on Y, which in turn depends on the distribution of the model at the specific time t.

Since moment-based analysis methods usually give approximations of the moments and not the exact values, we consider both, the demean estimator defined by Eq. (2.57) and the estimator of the two-step procedure in Eq. (3.4) for our numerical results in the following. In particular, if the theoretical moments are poorly approximated, it is likely that also the accuracy of the resulting estimates is poor.

3.3 Results

To analyze the performance of the GMM we consider two case studies, the simple gene expression model in Table 3.1 and a network of two genes with mutual repression, called exclusive switch [83]. The reactions of the exclusive switch are listed in Table 3.2. All propensities follow the law of mass action. For the parameters that we chose, the corresponding probability distribution is bi-modal.

For fixed reaction rate constants and initial conditions, we used the SSA to generate trajectories of the systems and record samples of the size of the corresponding protein/mRNA populations. In addition, we used the software tool SHAVE [81] to generate moment equations both for the standard moment closure and for the hybrid approach. In SHAVE the partial moments are integrated instead of the conditional moments such that the differential-algebraic equations are transformed into a system of (ordinary) differential equations after truncating modes with insignificant probabilities. Then an accurate approximation of the solution using standard numerical integration methods can be obtained. The system of moment equations is always closed by setting all central moments of order larger than m to zero. We used for the inference approach only the moments up to order $m-1$ since the precision of the moments of highest order m is often poor. SHAVE allows to export the (hybrid) moment equations as a MATLAB-compatible m-file. We then used MATLAB's ode45 solver, which is based on a fifth order Runge-Kutta method, to integrate the (hybrid) moment equations. Note that for the gene expression example, the moment equations are exact since all propensities are linear. Thus, even an analytic solution is possible for this system.

We then used MATLAB's Global Search routine to minimize the objective function in Eq. (2.51). Global Search is a method for finding the global minimum by starting a local solver from multiple starting points that are chosen according to a heuristic [123]. Therefore the total running time of our method depends on the tightness of the intervals that we use as constraints for the un-

Table 3.1: Simple gene expression model [110]: The evolution of the molecular populations DNA_{ON}, DNA_{OFF}, and mRNA is described by the random vector $\mathbf{X}(t)=(X_1(t),X_2(t),X_3(t))$, respectively.

Reactions	Propensities	Intervals
$DNA_{ON} \rightarrow DNA_{OFF}$	$\alpha_1(\mathbf{x}) = b \cdot x_1$	$b \in [0, 0.5]$
$DNA_{OFF} \rightarrow DNA_{ON}$	$\alpha_2(\mathbf{x}) = a \cdot x_2$	$a \in [0, 0.5]$
$DNA_{ON} \rightarrow DNA_{ON}+$ mRNA	$\alpha_3(\mathbf{x}) = c \cdot x_1$	$c \in [0, 0.5]$

Table 3.2: Exclusive switch model [83]: Two different proteins P_1 and P_2 can bind to a promoter region on the DNA. If P_1 is bound to the promoter the production of P_2 is inhibited and vice versa. In the free state both proteins can be produced.

Reactions, $i = 1, 2$	Rate constant	Interval
DNA $\rightarrow$ DNA + P_i	p_i	[0.5,1.5]
DNA.$P_i \rightarrow$ DNA.P_i + P_i	p_i	[0.5,1.5]
$P_i \rightarrow \emptyset$	d_i	[0,0.05]
DNA + $P_i \rightarrow$ DNA.P_i	b_i	[0,0.1]
DNA.$P_i \rightarrow$ DNA + P_i	u_i	[0,0.1]

known parameters as well as on the starting points of the Global Search procedure. The running times for one local solver call (using the hybrid approach for computing moments) were about 2 s (demean estimator) and 40 s (two-step estimator) for the gene expression model. For the exclusive switch, the average running time for a local solver call was about 2 min (demean) and 10 min (two-step). Note that the total running time depends on the amount of local solver calls carried out by Global Search, which varied between two and 50. For all experiments, we chose a single initial point that is located far away from the true values and allowed Global Search to choose 500 (potential) further starting points. Different initial points yielded similar results except if the initial points are chosen close to the true values (then the results are significantly better in particular in the case of only few moment constraints).

The intervals that we used as constraints for the parameters are all listed in Tables 3.1 and 3.2.

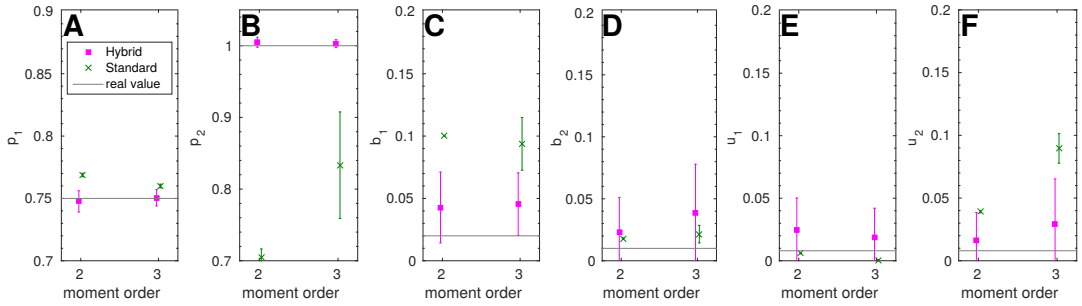


Figure 3.2: Exclusive switch model: Comparison of estimations with the demean procedure for the standard moment closure and hybrid moments.

3.3.1 Standard vs. Hybrid Moment-Based Analysis

In Fig. 3.2 we plot the results of a comparison between the standard and the hybrid moment closure when it is performed during the optimization procedure of the GMM inference approach. We chose the exclusive switch model for this since for this model the accuracy of the standard approach is poor. As an estimator for F, we used (2.57), which is based on demeaning (demean). Results for the two-step procedure show similar differences when standard and hybrid moment closure are compared. We fixed the degradation rates to ensure that identification of p_1 and p_2 is possible when the two protein populations are measured at only a single observation time point. To simultaneously identify all parameters (including p_1 and p_2) several observation time points are necessary (see Fig. 3.5).

The true values of the six unknown parameters are plotted against the means and standard deviations of the estimated values for a maximal moment order of 2 and 3, where for each of the six unknown parameters 50 estimations based on 10 000 samples each were used.

We see that the inaccurately approximated moments of the standard approach lead to severe problems in the inference approach. Nearly all parameters are estimated more accurately when the hybrid moment closure is used. For parameter b_1 most of the optimization runs converged to the upper limit of the given interval (0.1) when the standard approach was used. For the results in the sequel, we only used the hybrid moment closure.

3.3.2 Two-Step vs. Demean Approach

In Fig. 3.3 and 3.4 we plot results of the GMM approach applied to the two example networks, where we compare the performance of the two-step estimator

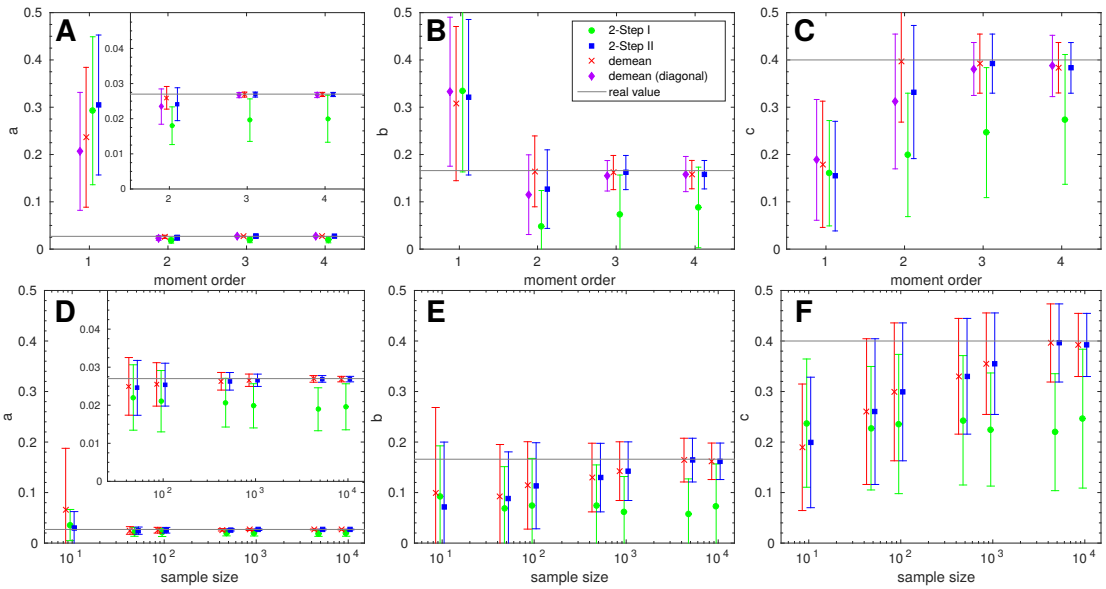


Figure 3.3: Gene expression model: Estimated parameters a,b and c for different numbers/orders of moments and $10\,000$ samples (**A**-**C**) and for different sample sizes based on 3 moments (**D**-**F**). The inner plots show results on a more detailed scale (**A** and **D**).

in Eq. (3.4) with the demean estimator in Eq. (2.57). We plot the true values of the parameters against the estimated values, where 2-Step I is the result of the first step of the two-step procedure (with $W = I$) and 2-Step II that of the second step (with $W = \hat{F}_1$ and $\hat{F}_1$ as defined in Eq. (3.4)).

For the results in Fig. 3.3 only one population (mRNA) was observed at $t = 100$ where the initial conditions were such that $\text{DNA}_{\text{OFF}} = 1$, $\text{DNA}_{\text{ON}} = 0$ and 10 mRNA molecules were present in the system. For three parameters the means and standard deviations of the estimated values are plotted, again based on 50 repetitions of the inference procedure.

In the first row of Fig. 3.3 the accuracy of the estimation is compared with respect to the number/order of moments considered, where again for each of the 50 estimated values $10\,000$ samples were used. We see that if only one moment is considered or if equal weights are used for the first two moments, only a rough estimate is possible since identification is not possible. The accuracy is markedly improved when the weights are chosen according to the demean approach. Here, it is important to note that for a maximal order of $m = 2$, in W we also consider, besides the squared cost functions $g_1(\theta)^2$ and $g_2(\theta)^2$, the mixed term $g_1(\theta)g_2(\theta)$. This additional term significantly improves the quality of the estimation such that it is possible to achieve a good estimation of the parameters

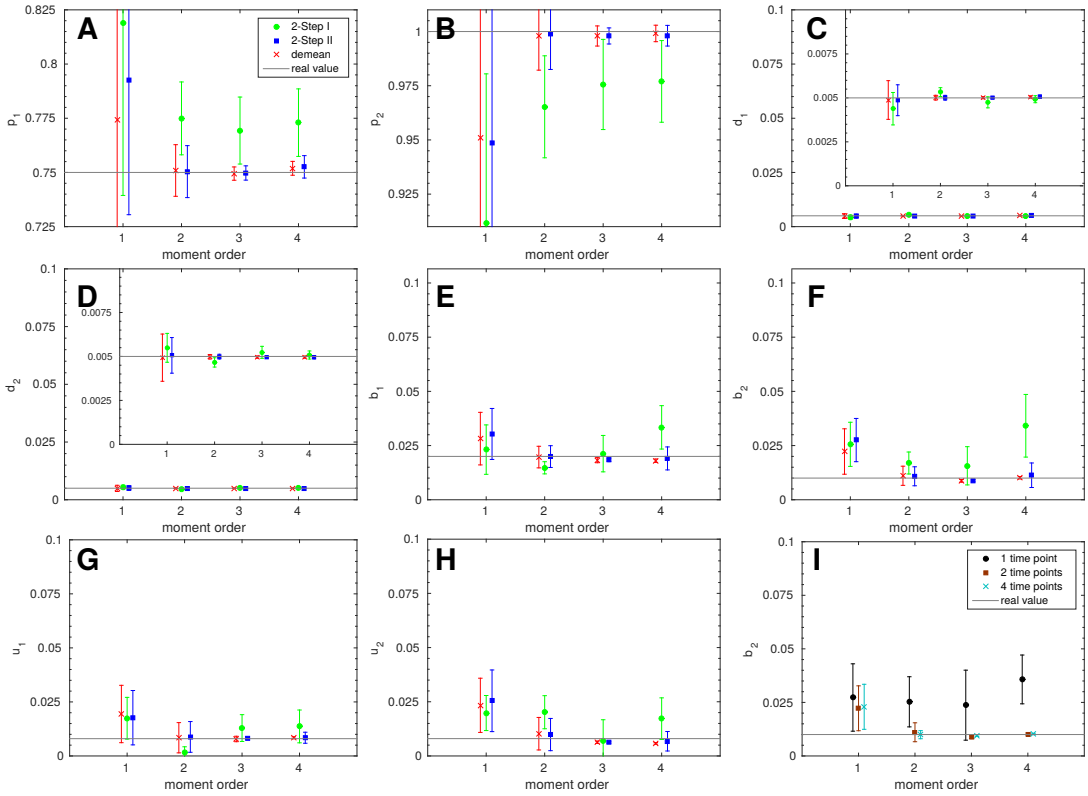


Figure 3.4: Exclusive switch model: Estimated parameters for maximal moment order 1-4 based on 10 000 independent samples observed at time $t = 100$ and $t = 200$ (**A**-**H**) and at 1-4 different time points for the demean-based estimation of b_2 (**I**). The inner plots show results on a more detailed scale (**C** and **D**).

with only the sample mean and the sample second moment. To further investigate the positive influence of the mixed term, we additionally plot results for the case that only variances are estimated, referred to as 'demean (diagonal)', i.e., the weight matrix is the inverse of a diagonal matrix that contains the variances estimated based on the demean approach.
However, the variance of the estimator for a maximum order of two is relatively high but decreases significantly when also the third (and fourth) moment is considered. Here, demean and the second step of the two-step procedure perform equally well and also demean (diagonal) gives very good results. Opposed to this $W = I$ (first step of two-step procedure) gives poor results and a high variance also if higher moments are considered.

In Table 3.3 we give an example for the (normalized) matrix W as used for demean and 2-Step II. The two methods choose nearly identical weights and the mean has the highest weight. Then, the mixed cost function for mean and sec-

Table 3.3: Weight matrices for the two-step and demean procedure with moment order 3 for the gene expression model. The entries are normalized with respect to the weight for the mean and rounded (the original weight matrices are both positive semi-definite).

method	W
Two-Step	$\begin{pmatrix} 1 & -0.0495 & 0.0007 \\ -0.0495 & 0.0025 & -3.86e^{-5} \\ 0.0007 & -3.86e^{-5} & 6.11e^{-7} \end{pmatrix}$
Demean	$\begin{pmatrix} 1 & -0.0494 & 0.0007 \\ -0.0494 & 0.0025 & -3.85e^{-5} \\ 0.0007 & -3.85e^{-5} & 6.09e^{-7} \end{pmatrix}$

ond moment has a (negative) weight of about $2 \cdot (-4.95)\%$ since these moments are negatively correlated (and so are the second and third moment). All terms that involve the third moment have a very small weight as their covariances are high.

It is important to note that also if the number of moment constraints m is equal to the number of parameters, q, 2-Step I performs poor (see results for maximal order $m = 3$ in the first row of Fig. 3.3). The reason is that in this example identification is not possible if only three terms are used due to functional dependencies between the parameters of the first two reactions and due to the fact that only at a single time point measurements were made. If identification was possible and the computed population moments were exact, the results should be independent of the choice of W for the case that q equals m.

Thus, the weights given by the estimators for F in (3.4) and (2.57) substantially increase the accuracy of the results and allow identification, because additional information about the covariances between the Y^r are used. Moreover, due to the off-diagonal entries of W additional mixed terms are part of the objective function.

In the second row in Fig. 3.3, we compare the accuracy for different samples sizes where the first three moments were considered. While 2-Step I does not show a systematic improvement when the number of samples increases, we see for 2-Step II and demean not only significantly improved estimates but also smaller variances. However, in the case of few samples, demean gives in particular for parameter a a high variance. This comes from the fact that the corresponding estimator uses the sample mean instead of the theoretical mean and therefore the weight matrix is far from optimal if N is small.

In Fig. 3.4, **A**-**H**, we plot results for the exclusive switch model where all eight parameters were estimated based on observations of the two protein populations of P_1 and P_2 at two time points. Recall that we need two time points to ensure identifiability of some parameters (see also Section 3.3.3). On the X axis the maximal order of moments used is plotted. For the orders 1, 2, 3 and 4 there are in total 2, 5, 9 or 14 moments, respectively. Again, 2-Step II and demean both give accurate results from a maximal order of two on, whereas 2-Step I gives poor results. In addition, the variance of the estimator decreases with increasing maximal order. However, the values for 2-Step II become slightly worse and have higher variance for a maximal order of four since these moments are not approximated very accurately. Also the accuracy of the demean estimator does not improve when the maximum order is increased from three to four. Thus, the cost functions of order four moments do not lead to any significant improvement in this example and should be excluded.

3.3.3 Multiple Time Points

For certain pairs of chemical reactions the identifying condition

$$E[\mathbf{f}(Y,\theta)] = \mathbf{0} \text{ if and only if } \theta = \theta_0.$$

is violated when only regarding snapshot data from a single time point. For example in the very simple reaction system

$$\begin{aligned} \emptyset &\to A \quad \text{rate } \lambda, \\ A &\to \emptyset \quad \text{rate } \mu \end{aligned}$$

every combination of λ and μ with $\frac{\lambda}{\mu} = const$ would lead to the same snapshot data for species A at a certain time point.

In order to resolve this problem more information, i.e. snapshot data at several time points (of independent samples to avoid correlation), is needed or one of the parameters has to be fixed. In Section 3.3.1 this problem already occurred for the exclusive switch: The corresponding rates are production p_i and degradation d_i as well as binding b_i and unbinding u_i. By fixing the degradation rates d_i the estimation of the production rates becomes quite well, whereas b_i and u_i can not be estimated due to the identifying problem.

For the following estimations the demean procedure was used. The two-step method showed a similar behavior. With no fixed parameters and only a single time point $t = 200$ nothing can be reliably estimated as indicated in Fig. 3.5. The estimated values are often far away from the real ones and the variance is also quite high in all cases.

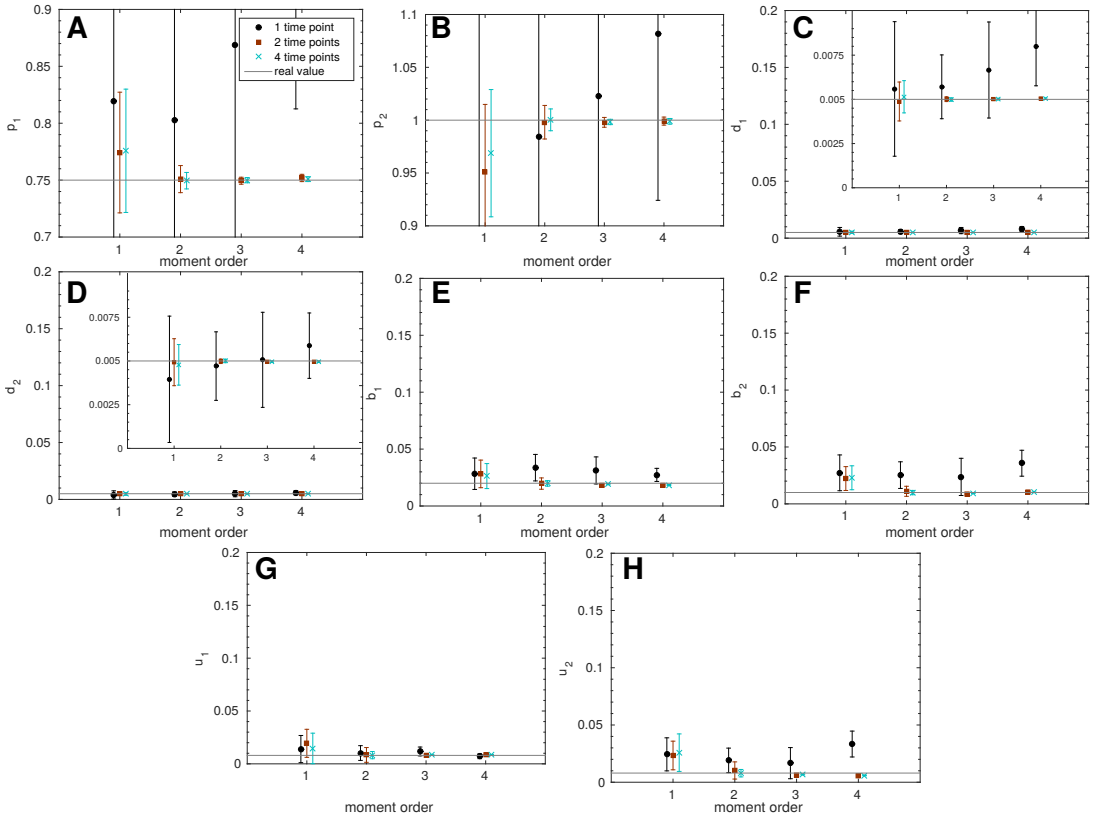


Figure 3.5: Exclusive switch model: Comparison of estimations with the demean procedure for single time point data and combined data for samples of two and four independent time points.

The consideration of a second time point ($t = 100, 200$) resolves the issue in case of sufficient moment conditions, i.e. order 2 or higher. Four time points ($t = 50, 100, 150, 200$) do not further improve the estimation but due to the higher total number of samples ($500\,000$ per time point) the variance is decreased.

3.3.4 Further Estimators

For our results we focused on the most popular GMM estimators, that is, demean and two-step. However, we also implemented two additional variants of estimators that were described before. One is the multi-step approach (or iterated GMM estimator) that results from further iterations of the two-step procedure. However, in our examples we did not see an increase in accuracy after the second iteration. Also, for the second approach,the continuously updating GMM estimator, the results did not show increased accuracy, even when we

used results of the other estimators (e.g. demean) as starting points for the optimization. Moreover, for large weight matrices, the recomputation in each step of the optimization resulted in longer running times.

Overall, our experiments show that for sufficiently large N the demean estimator usually yields the best results, while two-step performs better for small N. Moreover, choosing three as the maximum order gave the best results (accurate average value and small standard deviations) for the examples that we considered.

3.4 Discussions

In the context of stochastic chemical kinetics, parameter inference methods are either based on Markov chain Monte-Carlo (MCMC) schemes [17, 22, 43, 127], on approximate Bayesian computation techniques [26, 30, 122] or on maximum likelihood estimation using a direct approximation of the likelihood [110, 3] or a simulation-based estimate [121, 124]. Maximum likelihood estimators are, in a sense, the most informative estimates of unknown parameters [60] and have desirable mathematical properties such as unbiasedness, efficiency, and normality. On the other hand, the computational complexity of maximum likelihood estimation is high as it requires a simulation-based or numerical solution of the CME for many different parameter instances.

Since the applicability of these methods is limited, approaches based on moment closure [13, 34, 71, 97, 112, 128] or linear noise approximations [11, 70, 129] have been developed. An approximation of the likelihood of order-two sample moments is maximized in [13, 34, 112, 128]. The approach exploits that for large numbers of samples these sample moments are asymptotically normally distributed. The negative log-likelihood leads to an optimization problem where the differences between the sample and theoretical moments up to order two are weighted and minimized as well. As opposed to the GMM, the weight matrix in [112, 128] is estimated based on the theoretical moments of the model up to order four and independent of the samples while in the GMM approach this matrix depends on the samples (and theoretical moments up to order two). Moreover, the objective function contains an additional summand, which is the logarithm of the determinant of the estimated covariance matrix.

In [13], Bogomolov et al. insert sample instead of theoretical moments in the derived formulas for the covariances of moment conditions up to order two. A comparison for the two examples that we consider in the previous section yields that when the theoretical moments are used to estimate covariances, similar to the continuously updating GMM, optimization was slow and sometimes failed to return the global optimum due to a much more complex landscape of the

objective function. When sample moments are considered as suggested in [13], the results are similar to those of the GMM demean estimator for a maximum order of two. In [34], only variances are considered (weight matrix is diagonal) and estimated based on the samples. Therefore, it does not exploit the information contained in the mixed terms, which lead to improved estimates in our examples (see results for 'demean (diagonal)' in Fig. 3.3).

A similar approach is used in [97] where the moment equations are closed by a Gaussian approximation. The parameter estimation is based on using a ML estimator and a MCMC approach. In [71] the importance of higher moment orders when using least square estimators is shown. Weights for terms that correspond to different moments are chosen ad hoc and not based on any statistical framework. The GMM can also be used to estimate the parameters from the equilibrium distribution, where approximations such as moment closures are not necessary [9].

Here, we present results for the general method of moments that assigns optimal weights to the different moment conditions for an arbitrary maximal moment order and number of species. We showed that trivial weights (e.g. identity matrix) give results whose accuracy can be strongly increased when optimal weights are chosen. In the very common case that functional dependencies between parameters exist (e.g. degradation and production of the same species) and identification is difficult, the GMM estimator allows to accurately identify the parameters. Moreover, our results indicate that the accuracy of the estimation increases when moments of order higher than two are included. A general strategy could be to start with $m = q$ cost functions (equal to the number of unknown parameters) and increase the maximal order until tests for over-identifying restrictions (e.g. the Hansen test [52]) suggest that higher orders do not lead to an improvement. In this way, cost functions that do not improve the quality of the estimation, such as the fourth order cost functions for the results in Fig. 3.4, can be identified.

We also found that an accurate approximation of the moments is crucial for the performance of the GMM estimator. Thus, hybrid approaches such as the method of conditional moments [55] or sophisticated closure schemes (e.g. [13]) should be preferred. If all propensities in the network are linear, the moment equations are exact and model misspecification is not an issue. However, for most networks the moments can only be approximated, since the propensities are nonlinear, and hence the model is potentially misspecified. Again, statistical tests can be used to detect model misspecification [49] and equations for higher order moments may be added to the (conditional) moment equations to improve the approximation of the lower order moments.
Finally, we note that the GMM framework can also be applied when the ob-

served molecular counts are subject to measurement errors. It is straight forward to extend the GMM framework to the case of samples $Y_k + \varepsilon$ where the error term ε is independent and normally distributed with mean zero.

3.5 Conclusion

Parameter inference for stochastic models of cellular processes demands huge computational resources. The proposed generalized method of moments (GMM) approach is based on an adjustment of the statistical moments of the model and therefore does not require the computation of likelihoods. This makes the approach appealing for complex networks where stochastic effects play an important role, since the integration of the moment equations is typically fast compared to other computations such as the computation of likelihoods. The method does not make any assumptions about the distribution of the process (e.g. Gaussian) and complements the existing moment-based analysis approaches in a natural way.

Here, we used a multistart gradient-based minimization scheme, but the approach can be combined with any global optimization method. We found that the weights of the cost functions computed by the GMM estimator yield clearly more accurate results than trivial (identical) weights. In particular, the variance of the estimator decreases when moments of higher order are considered. We focused on the estimation of reaction rate constants and, as future work, we plan to investigate how well Hill coefficients and initial conditions are estimated.

An important advantage of the proposed method is that in the economics literature the properties of GMM estimators have been investigated in detail over decades and several variants and related statistical tests are available. We will also check how accurate approximations for the variance of the GMM estimator are [49]. Since we found that when moments of order higher than three are included, the results become slightly worse, we will also explore the usefulness of statistical tests for over-identifying moment conditions. In this way, we can ensure that only moments conditions are included that improve the estimation.

Chapter 4

Summary

To top this thesis off, we summarize the most important results from both Chapter 2 and Chapter 3 again.

In Chapter 2 we introduced a generalized hidden Markov model, that is able to model the methylation dynamics of whole sequences of CpGs simultaneously, in contrast existing models that usually consider only a single CpG. The inclusion of whole patterns enables us to investigate the possible influence of neighboring methylation states on methylation events and take possible correlations between the CpGs into account. Furthermore, since the CpGs do not behave independently anymore, it is possible to test different hypotheses on the working mechanisms of the methylation enzymes (Dnmts), since the order of methylation events matters. We show how to formally generate the transition probability matrix for whole CpG sequences by using a stochastic automata networks description with functional transitions. These functional transitions depend on the newly introduced dependence parameters and the methylation states of the adjacent CpGs.

To fit the model to biological data, we test different parameter estimation methods. For a small number of CpGs a maximum likelihood approach is viable, while for a larger number of CpGs we have to resort to the generalized method of moments (GMM) or a Bayesian approach. Our main biological findings are that Dnmt1 works processively and independent of the neighborhood, while Dnmt3a/b shows a dependence on the methylation state of the left neighbor. From whole genome data we get the result that CpGs in hypomethylated promoter regions behave independently, while hypermethylated CpGs from other regions show a dependence on the methylation state of the left neighbor, but not on the right.

In Chapter 3 we introduced the GMM, a moment-based parameter estimation technique from econometrics to chemical reaction networks. The main idea is that the moments are weighted based on their covariance. This has the two

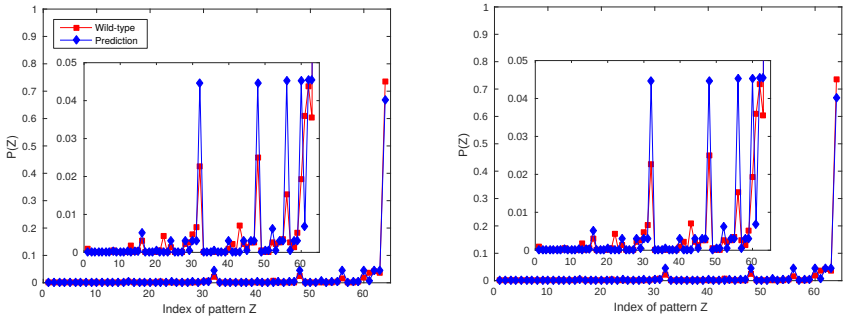
advantages, namely that the influence of moments of higher order (which are usually much larger in their order of magnitude and have a higher variance) is mitigated, such that they do not dominate the moments of low order, and that due to the non-diagonal entries in the weight matrix, more information can be included in the estimation process. Compared to trivial (identical) weights for all moments, we observe a clear improvement of the estimation qualitity for the same number of moments used. In particular, the GMM yields smaller variances for the estimated parameters, when moments of higher orders are used. Furthermore, if moment approximation techniques (like moment closure) have to be used, it is crucial to use to best approximation possible. We found that hybrid approaches yielded much better results in the parameter estimation than standard approximations.

Appendix A

Additional Figures

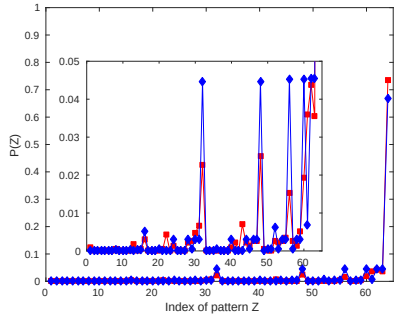
Here, we show some additional figures. In particular we show the comparison of the distribution from real wild-type data and the predicted distribution from our model (Eq.(2.28)). Since for each enzyme we have four possible models (Eqs.(2.23)-(2.26); models 1-4) for each of the two enzymes, there are $4^2 = 16$ possible models in total for the wild-type.

In Fig. A.1 we show all 16 results, where (x, y) means that we used model x for Dnmt1 and model y for Dnmt3a/b. The red symbols represent the distribution from the wild-type data and is hence identical in all 16 subplots. The blue symbols show the results from the numerical solution of the respective model. Note that in all cases there are only some small deviations on the large scale. In the insets we show a zoomed in version of all states with small probabilities, i.e., all except the fully methylated state 64. In absolute terms, even in the zoomed in version the deviations are rather small. Importantly, the models correctly predict the position of the peaks, i.e., correctly identifies more common patterns. Furthermore, the deviations between the predictions of the different models are only very subtle.

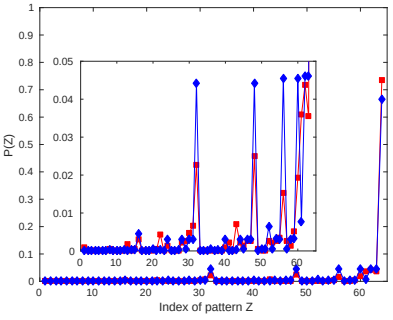


(a) (1,1)

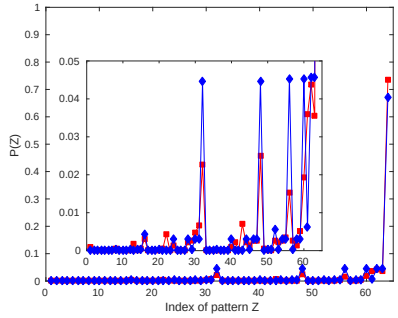
(b) (1,2)



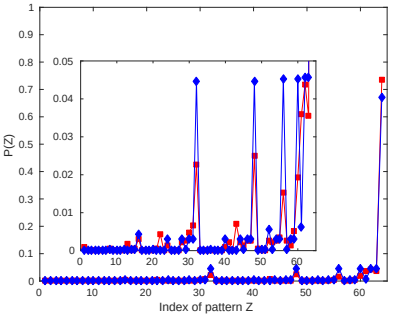
(c) (1,3)



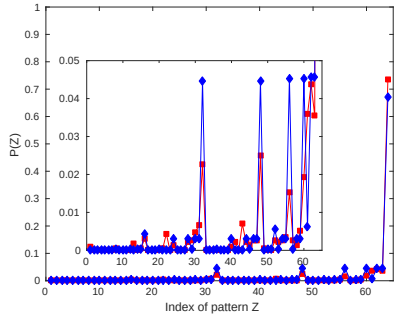
(d) (1,4)



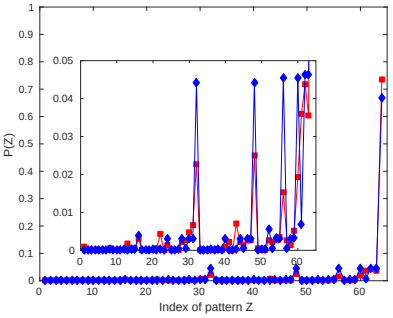
(e) (2,1)



(f) (2,2)

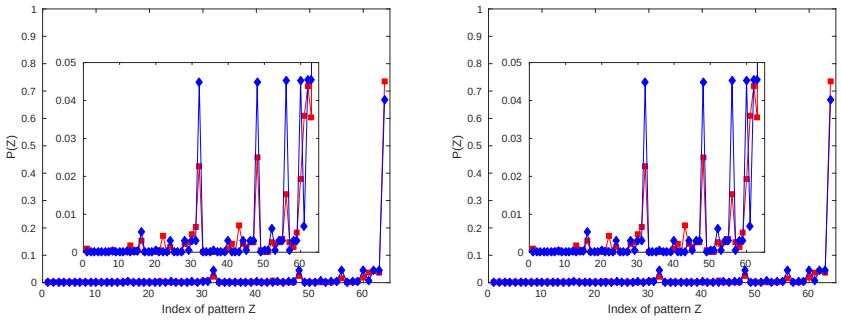


(g) (2,3)

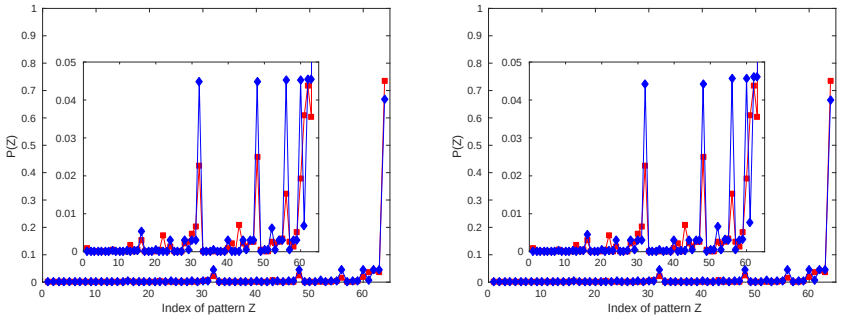


(h) (2,4)

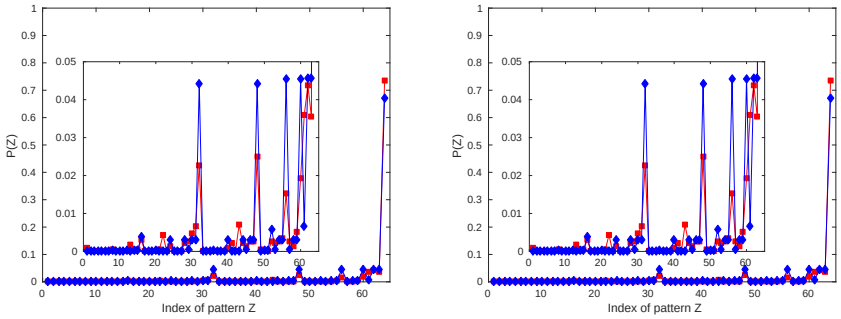
Figure A.1: The figures show the predicted and the measured pattern distribution for all 16 models for mSat. The inset shows a zoomed in version of the distribution.



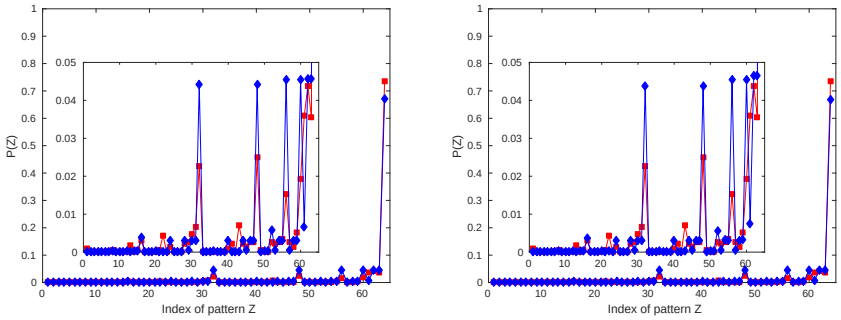
(i) (3,1) (j) (3,2)



(k) (3,3) (l) (3,4)



(m) (4,1) (n) (4,2)



(o) (4,3) (p) (4,4)

Figure A.1: (cont.) The figures show the predicted and the measured pattern distribution for all 16 models for mSat. The inset shows a zoomed in version of the distribution.

Appendix B

Pseudo Code

B.1 Stochastic Automata Networks

We now present the algorithm B.1 that is used to generate the transition matrix M in Section 2.3.5 for maintenance or *de novo* for L CpGs, given the respective transition matrix for a single CpG A and a function f that introduces the neighborhood dependency. Note that for cell division a standard Kronecker product can be applied since all CpGs behave independently in this case.

Since A (and hence M) is usually sparse, we initialize M as a matrix of containing only zeros (line 2) with the goal to only update the entries that will eventually be non-zero. The size of the matrix is determined by the number of possible states (4 in our example) and the number of CpGs L. We extract the indices of non-zero elements in A (line 3) and safe them to a list (line 4). Then a copy of this list is made (line 5), which will be updated later, while the list with single CpGs indices is kept, since it is needed to iteratively recompute the new indices for the larger matrices in the while loop (lines 8-19). In this loop the new indices are calculated with Eq. (2.34) (lines 13 and 14) and stored in a list (line 15). After calculating all new indices the old list of indices is replaced by the new one (line 18). This procedure repeats until the number of necessary Kronecker multiplications is reached.

Once all indices of the final matrix that contain non-zero elements are obtained, we calculate the corresponding entries by iterating over the indices (for loop in lines 21-31). First of all, the indices can be uniquely converted into an ordered list, such that we know the states of all CpGs before (from u) and after transition (from v) for a given pair of indices (lines 22 and 23). We then simply multiply the correct entries of the single CpG matrix A, where the indices stem from the ordered list (for loop in lines 25-30). For example for the first CpG, we use the first entries in the lists resulting from u and v and so on. If the en-

try contains the function f (line 27), we chose the correct transition probability based on the process (maintenance or *de novo*), the position (boundary or non-boundary) and assumption (e.g. processivity). The states of the adjacent CpGs can be obtained from the lists generated in lines 22 and 23.

Algorithm B.1 Generation of the transition matrix M of one of the subprocesses for L CpGs using SANs.

```
1:  procedure SANs(A, f, L)
2:      M ← zeros(numstate**L, numstate**L)
3:      i, j ← nonzero(A)
4:      ind1 ← list([i, j])
5:      indices ← ind1
6:      n ← 1
7:
8:      while n < L do
9:          n ← n + 1
10:         ind2 ← [ ]
11:         for i, j in ind1 do
12:             for u, v in indices do
13:                 u2 ← numstate * (u − 1) + i
14:                 v2 ← numstate * (v − 1) + j
15:                 ind2 ← ind2.append([u2, v2])
16:             end for
17:         end for
18:         indices ← ind2
19:     end while
20:
21:     for u, v in indices do
22:         indu ← Fragment(u)
23:         indv ← Fragment(v)
24:         M[u, v] ← 1
25:         for n = 1, . . . , L do
26:             if A[indu[n], indv[n]] is callable then
27:                 A[indu[n], indv[n]] ← f
28:             end if
29:             M[u, v] ← M[u, v] * A[indu[n], indv[n]]
30:         end for
31:     end for
32:     return M
33: end procedure
```

B.2 Approximate Bayesian Computation

Here we describe the algorithm B.2 of the simple version of the ABC-SMC that was introduced in Section 2.4.2. Note that for convenience the parameters in this pseudo-code are treated as univariate, however, the multivariate case works in the same fashion.

Given the distribution of the observed data $distr1$ and the desired size of the posterior n, the algorithm returns an approximation of the posterior distribution. At first we initialize an empty list (line 2) which is later used to store the posterior. We then perform a loop of size N_1 (lines 3-8) which is used to roughly scan the parameter space. In each iteration we draw a random parameter, which is uniformly distributed between 0 and 1 (line 4). We then obtain a distribution based on this parameter from either solving the model numerically (for few CpGs) or via Monte-Carlo simulation (for many CpGs) (line 6). Then a suitable measure is used (Euclidean distance, Hellinger distance, Kullback-Leibler, ...) to calculate the distance between observed and generated distribution (line 6). Finally, the parameter and the corresponding distance is saved to the list for the posterior (line 7). After finishing the scanning, the entries in the list are sorted with respect to the distance (line 9) and only the n best entries, i.e., entries with the smallest distance, are kept (line 10). Based on the remaining entries, the tolerance is calculated with Eq. (2.59) (line 11) and the weights are calculated with Eq. (2.60) (line 12).

We then try to improve the posterior by searching more thoroughly in parts of the parameter space that yielded good results while scanning the entire parameter space (for loop in lines 14-25). Instead of drawing uniformly distributed parameters, we now draw from a normal distribution (line 16), where the mean is a random value from the current posterior. We make a weighted random choice based on the weights that were calculated in line 12. The higher the weight (or smaller the distance) the more likely the parameter will be chosen as mean for the normal distribution (line 15). For the standard deviation σ we impose a small constant, which is small enough to reduce excessive rejections of parameters and large enough to allow for further exploration of the parameter space. Note that there are more sophisticated choices for σ, which also depend on the current posterior, like twice the weighted empirical variance of the current posterior as in [10]. Also note, that the parameters still have to be bound between 0 and 1. With a normal distribution that is not necessarily the case. For convenience sake, we omit the corresponding if query here. Similar to the first for loop, we now obtain a distribution for the drawn parameter from the model (line 17) and calculate the distance to the distribution from the real data (line 18). If the distance is smaller than the tolerance ε (line 19), we accept the parameter and the bottom (worst) entry in the posterior is replaced (line 20).

Then the list is sorted again (line 21) and the tolerance (line 22) and weights are recalculated (line 23).

Note that instead of the for loop with a fixed size (line 14), there are other criteria that may be used to terminate the algorithm when improving the posterior. Some possible choices are the maximal number of consecutive rejections or the minimal (relative) change in the tolerance after accepting a new parameter for the posterior.

Algorithm B.2 ABC-SMC

```
 1: procedure ABC(distr1, n)
 2:     post ← [ ]
 3:     for i = 1, ..., N_1 do
 4:         r ∼ unif(0, 1)
 5:         distr2 ← SolveModel(r)
 6:         d ← Distance(distr1, distr2)
 7:         post ← post.append([r, d])
 8:     end for
 9:     post ← Sort(post, d)
10:     post ← post[1 : n]
11:     ε ← CalculateEps(post)
12:     w ← Weights(post)
13:
14:     for i = 1, ..., N_2 do
15:         μ ← WeightedRandomChoice(post, w)
16:         r ∼ normal(μ, σ)
17:         distr2 ← SolveModel(r)
18:         d ← Distance(distr1, distr2)
19:         if d < ε then
20:             post[−1] ← [r, d]
21:             post ← Sort(post, d)
22:             ε ← CalculateEps(post)
23:             w ← Weights(post)
24:         end if
25:     end for
26:     return post
27: end procedure
```

Abbreviations

5caC	5-carboxylcytosine
5fC	5-formylcytosine
5hmC	5-hydroxymethylcytosine
5mC	5-methylcytosine
ABC	approximate Bayesian computation
ABC-SMC	ABC-sequential-Monte-Carlo
BS-seq	bisulfite sequencing
C	cytosine
CGI	CpG island
CME	chemical master equation
CpG	C and G linked via phosphate group
CTMC	continuous time Markov chain
Dnmt	DNA methyltransferase
DTMC	discrete time Markov chain
G	guanine
GMM	generalized method of moments
HMM	hidden Markov model
KO	knockout
MC	Monte-Carlo
MLE	maximum likelihood estimator
mRNA	messenger RNA
RV	random variable
SAN	stochastic automata network
SSA	stochastic simulation algorithm
WT	wild-type

List of Figures

List of Tables

List of Algorithms

Bibliography

[1] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Molecular Biology of the Cell*. Garland Science, 4th edition, 2002.

[2] Angelique Ale, Paul Kirk, and Michael PH Stumpf. A general moment expansion method for stochastic kinetic models. *J. Chem. Phys.*, 138(17):174101, 2013.

[3] Aleksandr Andreychenko, Linar Mikeev, David Spieler, and Verena Wolf. Approximate maximum likelihood estimation for stochastic chemical kinetics. *EURASIP J. Bioinform. Syst. Biol.*, 9, 2012.

[4] Alexander Andreychenko, Linar Mikeev, and Verena Wolf. Model Reconstruction for Moment-Based Stochastic Chemical Kinetics. *ACM TOMACS*, 25(2):1–19, 2015.

[5] Christof Angermueller, Heather J Lee, Wolf Reik, and Oliver Stegle. DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning. *Genome Biology*, 18(1):67, 2017.

[6] Julia Arand, David Spieler, Tommy Karius, Miguel R Branco, Daniela Meilinger, Alexander Meissner, Thomas Jenuwein, Guoliang Xu, Heinrich Leonhardt, Verena Wolf, et al. In vivo control of CpG and non-CpG DNA methylation by DNA methyltransferases. *PLoS Genet*, 8(6):e1002750, 2012.

[7] Chikashi Arita, Jonas Bosche, Alexander Lück, and Ludger Santen. Localization of a microtubule organizing center by kinesin motors. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(12):123210, 2017.

[8] Chikashi Arita, Alexander Lück, and Ludger Santen. Length regulation of microtubules by molecular motors: exact solution and density profiles. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(6):P06027, 2015.

[9] Michael Backenköhler, Luca Bortolussi, and Verena Wolf. Generalized method of moments for stochastic reaction networks in equilibrium. In *International Conference on Computational Methods in Systems Biology*, pages 15–29. Springer, 2016.

[10] Mark A Beaumont. Approximate Bayesian Computation in Evolution and Ecology. *Annual Review of Ecology, Evolution, and Systematics*, 41(1):379–406, 2010.

[11] Frank T Bergmann, Sven Sahle, and Christoph Zimmer. Piecewise parameter estimation for stochastic models in COPASI. *Bioinformatics*, page btv759, 2016.

[12] Timothy H Bestor and Vernon M Ingram. Two DNA methyltransferases from murine erythroleukemia cells: purification, sequence specificity, and mode of interaction with DNA. *Proceedings of the National Academy of Sciences*, 80(18):5559–5563, 1983.

[13] Sergiy Bogomolov, Thomas A Henzinger, Andreas Podelski, Jakob Ruess, and Christian Schilling. Adaptive moment closure for parameter inference of biochemical reaction networks. In *Proc. of CMSB'15*, pages 77–89. Springer International Publishing.

[14] Nicolas Bonello, James Sampson, John Burn, Ian J Wilson, Gail McGrown, Geoff P Margison, Mary Thorncroft, Philip Crossbie, Andrew C Povey, Mauro Santibanez-Koref, et al. Bayesian inference supports a location and neighbour-dependent model of DNA methylation propagation at the MGMT gene promoter in lung tumours. *Journal of Theoretical Biology*, 336:87–95, 2013.

[15] Michael J Booth, Miguel R Branco, Gabriella Ficz, David Oxley, Felix Krueger, Wolf Reik, and Shankar Balasubramanian. Quantitative sequencing of 5-methylcytosine and 5-hydroxymethylcytosine at single-base resolution. *Science*, 336(6083):934–937, 2012.

[16] Michael J Booth, Tobias WB Ost, Dario Beraldi, Neil M Bell, Miguel R Branco, Wolf Reik, and Shankar Balasubramanian. Oxidative bisulfite sequencing of 5-methylcytosine and 5-hydroxymethylcytosine. *Nature Protocols*, 8(10):1841, 2013.

[17] Richard J Boys, Darren J Wilkinson, and Thomas BL Kirkwood. Bayesian inference for a discretely observed stochastic kinetic model. *Stat. Comp.*, 18:125–135, 2008.

[18] Peter Buchholz. Equivalence relations for stochastic automata networks. In *Computations with Markov Chains*, pages 197–215. Springer, 1995.

[19] Peter Buchholz. Hierarchical Markovian models: symmetries and reduction. *Performance Evaluation*, 22(1):93–110, 1995.

[20] Peter Buchholz and Peter Kemper. Kronecker based matrix representations for large Markov models. In *Validation of Stochastic Systems*, pages 256–295. Springer, 2004.

[21] Luis Busto-Moner, Julien Morival, Honglei Ren, Arjang Fahim, Zachary Reitz, Timothy L Downing, and Elizabeth L Read. Stochastic modeling reveals kinetic heterogeneity in post-replication DNA methylation. *PLOS Computational Biology*, 16(4):e1007195, 2020.

[22] Bernie J Daigle, Min K Roh, Linda R Petzold, and Jarad Niemi. Accelerated maximum likelihood parameter estimation for stochastic biochemical systems. *BMC Bioinformatics*, 13(1):68, 2012.

[23] Partha M Das and Rakesh Singal. DNA methylation and cancer. *Journal of Clinical Oncology*, 22(22):4632–4642, 2004.

[24] David L Davies and Donald W Bouldin. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

[25] Marc Davio. Kronecker products and shuffle algebra. *IEEE Transactions on Computers*, 100(2):116–125, 1981.

[26] Christopher C Drovandi and Anthony N Pettitt. Estimation of parameters for macroparasite population evolution using approximate Bayesian computation. *Biometrics*, 67(1):225–233, 2011.

[27] Melanie Ehrlich, Miguel A Gama-Sosa, Lan-Hsiang Huang, Rose Marie Midgett, Kenneth C Kuo, Roy A McCune, and Charles Gehrke. Amount and distribution of 5-methylcytosine in human DNA from different types of tissues or cells. *Nucleic acids research*, 10(8):2709–2721, 1982.

[28] Max Emperle, Arumugam Rajavelu, Richard Reinhardt, Renata Z Jurkowska, and Albert Jeltsch. Cooperative DNA binding and protein/DNA fiber formation increases the activity of the Dnmt3a DNA methyltransferase. *Journal of Biological Chemistry*, 289(43):29602–29613, 2014.

[29] Stefan Engblom. Computing the moments of high dimensional solutions of the master equation. *Appl. Math. Comput.*, 180(2):498–515, 2006.

[30] Paul Fearnhead and Dennis Prangle. Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. *J. R. Stat. Soc. Series B Stat. Methodol.*, 74(3):419–474, 2012.

[31] Suhua Feng, Shawn J Cokus, Xiaoyu Zhang, Pao-Yang Chen, Magnolia Bostick, Mary G Goll, Jonathan Hetzel, Jayati Jain, Steven H Strauss, Marnie E Halpern, et al. Conservation and divergence of methylation patterning in plants and animals. *Proceedings of the National Academy of Sciences*, 107(19):8689–8694, 2010.

[32] Paulo Fernandes, Brigitte Plateau, and William J Stewart. Efficient Descriptor-Vector Multiplications in Stochastic Automata Networks. *J. ACM*, 45(3):381–414, May 1998.

[33] John T Finch and Aaron Klug. Solenoidal model for superstructure in chromatin. *Proceedings of the National Academy of Sciences*, 73(6):1897–1901, 1976.

[34] Fabian Fröhlich, Philipp Thomas, Atefeh Kazeroonian, Fabian J Theis, Ramon Grima, and Jan Hasenauer. Inference for stochastic chemical kinetics using moment equations and system size expansion. *PLoS Comput Biol*, 12(7):1–28, 07 2016.

[35] Audrey Qiuyan Fu, Diane P Genereux, Reinhard Stöger, Charles D Laird, and Matthew Stephens. Statistical inference of transmission fidelity of DNA methylation patterns over somatic cell divisions in mammals. *The Annals of Applied Statistics*, 4(2):871, 2010.

[36] Diane P Genereux, Brooks E Miner, Carl T Bergstrom, and Charles D Laird. A population-epigenetic model to infer site-specific methylation rates from double-stranded DNA methylation patterns. *PNAS*, 102(16):5802–5807, 2005.

[37] Pascal Giehr. The role of Dnmts and Tets in shaping the DNA methylation landscape of mouse embryonic stem cells. 2019.

[38] Pascal Giehr, Charalampos Kyriakopoulos, Gabriella Ficz, Verena Wolf, and Jörn Walter. The Influence of Hydroxylation on Maintaining CpG Methylation Patterns: A Hidden Markov Model Approach. *PLoS Comput Biol*, 12(5):e1004905, 2016.

[39] Pascal Giehr, Charalampos Kyriakopoulos, Konstantin Lepikhov, Stefan Wallner, Verena Wolf, and Jörn Walter. Two are better than one: HPoxBS-hairpin oxidative bisulfite sequencing. *Nucleic Acids Research*, 46(15):e88–e88, 2018.

[40] Pascal Giehr and Jörn Walter. Hairpin bisulfite sequencing: synchronous methylation analysis on complementary DNA strands of individual chromosomes. In *DNA Methylation Protocols*, pages 573–586. Springer, 2018.

[41] Daniel T Gillespie. A general method for numerically simulating the time evolution of coupled chemical reactions. *J. Comput. Phys.*, 22:403–434, 1976.

[42] Daniel T Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25):2340–2361, 1977.

[43] Andrew Golightly and Darren J Wilkinson. Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte Carlo. *Interface focus*, page rsfs20110047, 2011.

[44] Humaira Gowher and Albert Jeltsch. Molecular enzymology of the catalytic domains of the Dnmt3a and Dnmt3b DNA methyltransferases. *Journal of Biological Chemistry*, 277(23):20409–20414, 2002.

[45] Winfried K Grassmann. *Computational probability*, volume 24. Springer Science & Business Media, 2013.

[46] Ramon Grima. A study of the accuracy of moment-closure approximations for stochastic chemical kinetics. *J. Chem. Phys.*, 136(15):154105, 2012.

[47] Konrad Grosser and Dirk Metzler. Modeling methylation dynamics with simultaneous changes in CpG islands. *BMC Bioinformatics*, 21(1):1–13, 2020.

[48] Jan O Haerter, Cecilia Lövkvist, Ian B Dodd, and Kim Sneppen. Collaboration between CpG sites is needed for stable somatic inheritance of DNA methylation states. *Nucleic Acids Research*, 42(4):2235–2244, 2013.

[49] Alastair R Hall. *Generalized Method of Moments*. Advanced Texts in Econometrics. OUP Oxford, , 2004.

[50] Alastair R Hall et al. *Generalized method of moments*. Oxford University Press Oxford, , 2005.

[51] Mary Beth Hanley, Woodrow Lomas, Dev Mittar, Vernon Maino, and Emily Park. Detection of low abundance RNA molecules in individual cells by flow cytometry. *PLoS ONE*, 8(2):1–8, 02 2013.

[52] Lars Peter Hansen. Large sample properties of generalized method of moments estimators. *Econometrica*, pages 1029–1054, 1982.

[53] Lars Peter Hansen, John Heaton, and Amir Yaron. Finite-sample properties of some alternative GMM estimators. *J. Bus. Econ. Stat.*, 14(3):262–280, 1996.

[54] Jan Hasenauer, Steffen Waldherr, Malgorzata Doszczak, Nicole Radde, Peter Scheurich, and Frank Allgöwer. Identification of models of heterogeneous cell populations from population snapshot data. *BMC Bioinformatics*, 12(1):1–15, 2011.

[55] Jan Hasenauer, Verena Wolf, Atefeh Kazeroonian, and Fabian J Theis. Method of conditional moments (MCM) for the chemical master equation. *J. Math. Biol.*, 69(3):687–735, 2013.

[56] Andreas Hellander and Per Lötstedt. Hybrid method for the chemical master equation. *J. Comput. Phys.*, 227(1):100 – 122, 2007.

[57] Thomas A Henzinger, Maria Mateescu, Linar Mikeev, and Verena Wolf. Hybrid numerical solution of the chemical master equation. In *Proc. of CMSB'10*, , 2010. ACM DL.

[58] Thomas A Henzinger, Maria Mateescu, and Verena Wolf. Sliding window abstraction for infinite Markov chains. In *Proc. CAV*, volume 5643 of *LNCS*, , 2009. Springer.

[59] Andrea Hermann, Rachna Goyal, and Albert Jeltsch. The Dnmt1 DNA-(cytosine-c5)-methyltransferase methylates DNA processively with high preference for hemimethylated target sites. *Journal of Biological Chemistry*, 279(46):48350–48359, 2004.

[60] James J Higgins. Bayesian inference and the optimality of maximum likelihood estimation. *Int. Stat. Rev.*, 45(1):9–11, 1977.

[61] Celeste Holz-Schietinger and Norbert O Reich. The inherent processivity of the human de novo methyltransferase 3A (DNMT3A) is enhanced by DNMT3L. *Journal of Biological Chemistry*, 285(38):29091–29100, 2010.

[62] Jean Jacod and Philip Protter. *Probability essentials*. Springer Science & Business Media, 2012.

[63] Tobias Jahnke. On reduced models for the chemical master equation. *SIAM MMS*, 9(4):1646–1676, 2011.

[64] Garrett Jenkinson, Jordi Abante, Andrew P Feinberg, and John Goutsias. An information-theoretic approach to the modeling and analysis of whole-genome bisulfite sequencing data. *BMC Bioinformatics*, 19(1):87, 2018.

[65] Nicolaas Godfried Van Kampen. *Stochastic Processes in Physics and Chemistry.* Elsevier, 3rd edition, 2007.

[66] Chantriolnt-Andreas Kapourani and Guido Sanguinetti. Higher order methylation features for clustering and prediction in epigenomic studies. *Bioinformatics*, 32(17):i405–i412, 2016.

[67] Chantriolnt-Andreas Kapourani and Guido Sanguinetti. Melissa: Bayesian clustering and imputation of single-cell methylomes. *Genome Biology*, 20(1):61, 2019.

[68] Atefeh Kazeroonian, Fabian Fröhlich, Andreas Raue, Fabian J Theis, and Jan Hasenauer. CERENA: Chemical reaction network analyzer: A toolbox for the simulation and analysis of stochastic chemical kinetics. *PloS one*, 11(1):e0146732, 2016.

[69] Arne Klungland and Adam B Robertson. Oxidized C5-methyl cytosine bases in DNA: 5-hydroxymethylcytosine; 5-formylcytosine; and 5-carboxycytosine. *Free Radical Biology and Medicine*, 107:62–68, 2017.

[70] Michał Komorowski, Bärbel Finkenstädt, Claire V Harper, and David A Rand. Bayesian inference of biochemical kinetic parameters using the linear noise approximation. *BMC Bioinformatics*, 10(1):1–10, 2009.

[71] Philipp Kügler. Moment fitting for parameter inference in repeatedly and partially observed stochastic biological models. *PloS one*, 7(8):e43001, 2012.

[72] Mahesh Kumar and Nitin R Patel. Clustering data with measurement errors. *Computational Statistics & Data Analysis*, 51(12):6084 – 6101, 2007.

[73] Suresh Kumar, Viswanathan Chinnusamy, and Trilochan Mohapatra. Epigenetics of modified DNA bases: 5-methylcytosine and beyond. *Frontiers in genetics*, 9:640, 2018.

[74] Pavel Kurasov, Alexander Lück, Delio Mugnolo, and Verena Wolf. Stochastic hybrid models of gene regulatory networks–A PDE approach. *Mathematical biosciences*, 305:170–177, 2018.

[75] Jens Kurreck. RNA Interference: From Basic Research to Therapeutic Applications. *Angewandte Chemie International Edition*, 48(8):1378–1398, 2009.

[76] Charalampos Kyriakopoulos. Stochastic modeling of DNA demethylation dynamics in ESCs. 2019.

[77] Charalampos Kyriakopoulos, Pascal Giehr, Alexander Lück, Jörn Walter, and Verena Wolf. A Hybrid HMM Approach for the Dynamics of DNA Methylation. *arXiv preprint arXiv:1901.06286*, 2019.

[78] Charalampos Kyriakopoulos, Pascal Giehr, and Verena Wolf. H(O)TA: estimation of DNA methylation and hydroxylation levels and efficiencies from time course data. *Bioinformatics*, 33(11):1733–1734, 2017.

[79] Michelle R Lacey, Melanie Ehrlich, et al. Modeling dependence in methylation patterns with application to ovarian carcinomas. *Stat Appl Genet Mol Biol*, 8(1):40, 2009.

[80] Charles D Laird, Nicole D Pleasant, Aaron D Clark, Jessica L Sneeden, KM Anwarul Hassan, Nathan C Manley, Jay C Vary, Todd Morgan, R Scott Hansen, and Reinhard Stöger. Hairpin-bisulfite PCR: assessing epigenetic methylation patterns on complementary strands of individual DNA molecules. *PNAS*, 101(1):204–209, 2004.

[81] Maksim Lapin, Linar Mikeev, and Verena Wolf. SHAVE – Stochastic hybrid analysis of Markov population models. In *Proc. of HSCC'11*, ACM International Conference Proceeding Series, 2011.

[82] Gangning Liang, Matilda F Chan, Yoshitaka Tomigahara, Yvonne C Tsai, Felicidad A Gonzales, En Li, Peter W Laird, and Peter A Jones. Cooperativity between DNA methyltransferases in the maintenance methylation of repetitive elements. *Molecular and cellular biology*, 22(2):480–491, 2002.

[83] Adiel Loinger, Azi Lipshtat, Nathalie Q Balaban, and Ofer Biham. Stochastic simulations of genetic switch systems. *Phys. Rev. E*, 75:021904, 2007.

[84] Cecilia Lövkvist, Ian B Dodd, Kim Sneppen, and Jan O Haerter. DNA methylation in human epigenomes depends on local topology of CpG sites. *Nucleic Acids Research*, 44(11):5123–5132, 2016.

[85] Alexander Lück. Replicated Computational Results (RCR) Report for "Automatic Moment-Closure Approximation of Spatially Distributed Collective Adaptive Systems". *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 26(4):27, 2016.

[86] Alexander Lück, Pascal Giehr, Karl Nordström, Jörn Walter, and Verena Wolf. Hidden Markov modelling reveals neighborhood dependence of Dnmt3a and 3b activity. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2019.

[87] Alexander Lück, Pascal Giehr, Jörn Walter, and Verena Wolf. A Stochastic Model for the Formation of Spatial Methylation Patterns. In *International Conference on Computational Methods in Systems Biology*, pages 160–178. Springer, Cham, 2017.

[88] Alexander Lück and Verena Wolf. Generalized method of moments for estimating parameters of stochastic reaction networks. *BMC Systems Biology*, 10(1):98, 2016.

[89] Alexander Lück and Verena Wolf. Generalized Method of Moments Estimation for Stochastic Models of DNA Methylation Patterns. *arXiv preprint arXiv:1911.01174*, 2019.

[90] Alexander Lück and Verena Wolf. A Stochastic Automata Network Description for Spatial DNA-Methylation Models. In *International Conference on Measurement, Modelling and Evaluation of Computing Systems*, pages 54–64. Springer, 2020.

[91] Pavlo Lutsik, Lars Feuerbach, Julia Arand, Thomas Lengauer, Jörn Walter, and Christoph Bock. BiQ Analyzer HT: locus-specific analysis of DNA methylation by high-throughput bisulfite sequencing. *Nucleic Acids Research*, 39(suppl_2):W551–W556, 2011.

[92] Maria Mateescu, Verena Wolf, Frederic Didier, and Thomas A Henzinger. Fast adaptive uniformisation of the chemical master equation. *IET Syst. Biol.*, 4(6):441–452, 2010.

[93] Daniel McNeish. On using Bayesian methods to address small sample problems. *Structural Equation Modeling: A Multidisciplinary Journal*, 23(5):750–773, 2016.

[94] Donald A McQuarrie. Stochastic approach to chemical kinetics. *Journal of Applied Probability*, 4:413–478, 1967.

[95] Stephan Menz, Juan C Latorre, Christof Schutte, and Wilhelm Huisinga. Hybrid stochastic–deterministic solution of the chemical master equation. *SIAM MMS*, 10(4):1232–1262, 2012.

[96] Karlene Nicole Meyer and Michelle Lacey. Modeling Methylation Patterns with Long Read Sequencing Data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2017.

[97] Peter Milner, Colin S Gillespie, and Darren J Wilkinson. Moment closure based parameter inference of stochastic kinetic models. *Stat. Comput.*, 23(2):287–295, 2013.

[98] James R Morris. Genes, genetics, and epigenetics: a correspondence. *Science*, 293(5532):1103–1105, 2001.

[99] Brian Munsky, Zachary Fox, and Gregor Neuert. Integrating single-molecule experiments and discrete stochastic models to understand heterogeneous gene transcription dynamics. *Methods*, 85:12 – 21, 2015. Inferring Gene Regulatory Interactions from Quantitative High-Throughput Measurements.

[100] Brian Munsky and Mustafa Khammash. The finite state projection algorithm for the solution of the chemical master equation. *J. Chem. Phys.*, 124:044144, 2006.

[101] Tally Naveh-Many and Howard Cedar. Active gene sequences are undermethylated. *Proceedings of the National Academy of Sciences*, 78(7):4246–4250, 1981.

[102] Allison B Norvil, Christopher J Petell, Lama Alabdi, Lanchen Wu, Sandra Rossie, and Humaira Gowher. Dnmt3b Methylates DNA by a Noncooperative Mechanism, and Its Activity Is Unaffected by Manipulations at the Predicted Dimer Interface. *Biochemistry*, 2016.

[103] Masaki Okano, Daphne W Bell, Daniel A Haber, and En Li. DNA methyltransferases Dnmt3a and Dnmt3b are essential for de novo methylation and mammalian development. *Cell*, 99(3):247–257, 1999.

[104] Masaki Okano, Shaoping Xie, and En Li. Cloning and characterization of a family of novel mammalian DNA (cytosine-5) methyltransferases. *Nature Genetics*, 19(3):219–220, 1998.

[105] Sarah P Otto and Virginia Walbot. DNA methylation in eukaryotes: kinetics of demethylation and de novo methylation during the life cycle. *Genetics*, 124(2):429–437, 1990.

[106] Pierre-Yves Perche, Michel Robert-Nicoud, Saadi Khochbin, and Claire Vourc'h. Nucleosome differentiation: role of histone H2A variants. *Medecine Sciences : M/S*, 19(11):1137—1145, November 2003.

[107] Lawrence R Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[108] Aharon Razin. CpG methylation, chromatin structure and gene silencing—a three-way connection. *The EMBO journal*, 17(17):4905–4908, 1998.

[109] Aharon Razin, C Webb, Moshe Szyf, Joel Yisraeli, A Rosenthal, Tally Naveh-Many, N Sciaky-Gallili, and Howard Cedar. Variations in DNA methylation during mouse cell differentiation in vivo and in vitro. *Proceedings of the National Academy of Sciences*, 81(8):2275–2279, 1984.

[110] Stefan Reinker, Rachel M Altman, and Jens Timmer. Parameter estimation in stochastic biochemical reactions. *IEEE Proc. Syst. Biol*, 153:168–178, 2006.

[111] Jakob Ruess. Minimal moment equations for stochastic models of biochemical reaction networks with partially finite state space. *J. Chem. Phys.*, 143(24):244103, 2015.

[112] Jakob Ruess and John Lygeros. Moment-based methods for parameter inference and experiment design for stochastic biochemical reaction networks. *ACM TOMACS*, 25(2):8, 2015.

[113] David Schnoerr, Guido Sanguinetti, and Ramon Grima. Comparison of different moment-closure approximations for stochastic chemical kinetics. *J. Chem. Phys.*, 143(18):185101, 2015.

[114] Roger B Sidje, Kevin Burrage, and Shev MacNamara. Inexact uniformization method for computing transient distributions of Markov chains. *SIAM J. Sci. Comput.*, 29(6):2562–2580, 2007.

[115] Abhyudai Singh and Joao Pedro Hespanha. Lognormal moment closures for biochemical reactions. In *Decision and Control, 2006 45th IEEE Conference on*, pages 2063–2068. IEEE, 2006.

[116] Mohammad Soltani, Cesar Augusto Vargas-Garcia, and Abhyudai Singh. Conditional moment closure schemes for studying stochastic dynamics of genetic circuits. *IEEE Transactions on Biomedical Circuits and Systems*, 9(4):518–526, Aug 2015.

[117] Laura B Sontag, Matthew C Lorincz, and E Georg Luebeck. Dynamics, stability and inheritance of somatic DNA methylation imprints. *Journal of Theoretical Biology*, 242(4):890–899, 2006.

[118] William J Stewart, Karim Atif, and Brigette Plateau. The numerical solution of stochastic automata networks. *European Journal of Operational Research*, 86(3):503–525, 1995.

[119] Miho M Suzuki and Adrian Bird. DNA methylation landscapes: provocative insights from epigenomics. *Nature Reviews Genetics*, 9(6):465–476, 2008.

[120] Philipp Thomas and Ramon Grima. Approximate probability distributions of the master equation. *Phys. Rev. E*, 92(1):012120, 2015.

[121] Tianhai Tian, Songlin Xu, Junbin Gao, and Kevin Burrage. Simulated maximum likelihood method for estimating kinetic rates in gene expression. *Bioinformatics*, 23:84–91, 2007.

[122] Tina Toni, David Welch, Natalja Strelkowa, Andreas Ipsen, and Michael PH Stumpf. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, 6(31):187–202, 2009.

[123] Zsolt Ugray, Leon Lasdon, John Plummer, Fred Glover, James Kelly, and Rafael Martí. Scatter search and local nlp solvers: A multistart framework for global optimization. *INFORMS JOC*, 19(3):328–340, 2007.

[124] Bilge Uz, Erdem Arslan, and Ian J Laurenzi. Maximum likelihood estimation of the kinetics of receptor-mediated adhesion. *J. Theor. Biol.*, 262(3):478 – 487, 2010.

[125] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269, 1967.

[126] Conrad H Waddington. The epigenotype. *Endeavour*, 1:18–20, 1942.

[127] Darren J Wilkinson. *Stochastic modelling for systems biology*. CRC press, , 2011.

[128] Christoph Zechner, Jakob Ruess, Peter Krenn, Serge Pelet, Matthias Peter, John Lygeros, and Heinz Koeppl. Moment-based inference predicts bimodality in transient gene expression. *PNAS*, 109(21):8340–8345, 2012.

[129] Christoph Zimmer and Sven Sahle. Deterministic inference for stochastic systems using multiple shooting and a linear noise approximation for the transition probabilities. *Systems Biology, IET*, 9(5):181–192.