HELMHOLTZ ASSOCIATION
Alliance on Systems Biology
HelmholtzZentrum münchen
German Research Center for Environmental Health

TECHNISCHE UNIVERSITÄT MÜNCHEN

Stochastic and deterministic methods for the analysis of Nanog dynamics in mouse embryonic stem cells

Justin Shane Feigelman

October 2015

TECHNISCHE UNIVERSITÄT MÜNCHEN

Fakultät für Mathematik

Stochastic and deterministic methods for the analysis of Nanog dynamics in mouse embryonic stem cells

Justin Shane Feigelman

Vollständiger Abdruck der von der Fakultät für Mathematik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:

Univ.-Prof. C. Czado, Ph.D.

Prüfer der Dissertation:

1. Univ.-Prof. Dr. Dr. F. J. Theis
2. Univ.-Prof. Dr. I. F. Sbalzarini, Technische Universität Dresden

Die Dissertation wurde am 17.11.2015 bei der Technischen Universität München eingereicht und durch die Fakultät für Mathematik am 03.03.2016 angenommen.

Abstract

Mouse embryonic stem cells (mESCs) are a useful model for understanding the regulatory mechanisms underlying embryonic development and cellular pluripotency. A thorough understanding of the mechanisms controlling pluripotency would therefore afford valuable insight into directed differentiation protocols, with applications ranging from cancer therapies to organ regeneration. Thus, it is vital to achieve a detailed, mechanistic understanding of mESC regulation and expression dynamics.

At the heart of the mESC regulatory network is the homeodomain-containing transcription factor Nanog. Nanog over-expression prevents differentiation, while down-regulation increases risk of spontaneous conversion to the endodermal lineage. Furthermore, Nanog interacts with hundreds of validated pluripotency factors, establishing it as a central regulator of pluripotency. Interestingly, Nanog expression is heterogeneous in mESC colonies culture in serum/LIF. However, Nanog heterogeneity is poorly characterized, largely due to lack of Nanog protein reporters.

In this thesis, I characterize the behavior of mESC colonies using a fluorescent fusion protein reporter developed by collaborators. I describe the expression dynamics of single cells in terms of transitions between mother and daughter cells and onsets of Nanog protein production from low-sorted subpopulations. I further investigate the possibility of oscillations previously hypothesized to give rise to the observed Nanog population-level heterogeneity. Using the single-cell time series, we identify a novel, subpopulation of mESCs with persistently low expression of Nanog, that yet remain pluripotent. I characterize the mosaic and low Nanog expression subpopulations in terms of their Pearson and partial correlations, and find that they exhibit unique correlation structures.

These analyses are complemented by the development of a tool for the investigation of local correlation structures in low-dimensional gene expression datasets, dubbed Multiresolution Correlation Analysis, that is applied to data obtained from mESC colonies that have been stained with fluorescent antibodies against the pluripotency factors Oct4, Sox2 and Klf4. Using this tool we find evidence for differential regulation between Nanog low/negative and Nanog mosaic colonies.

I investigate stochastic models for describing Nanog expression and regulation. To this end, I implemented and tested a novel analytical approximation developed by collaborators, for the purpose of inferring model parameters in a simple two-stage model of gene expression. Surprisingly, however, the approximation proved not useful for parameter inference due to the occurrence of non-physical negative transition densities.

Lastly, I implement an efficient exact Bayesian parameter inference algorithm adapted to inference for fully stochastic chemical reaction network models. I extend this method

to make it suitable for inference in colonies of proliferating cells. I demonstrate the utility of the method via testing with simulated data, and show that it is able to consistently identify the correct model topology when tested on data generated from models with either negative, positive or no transcriptional feedback control. I then apply the algorithm to a real cellular genealogy obtained via a Nanog fluorescent reporter. I present results for parameter inference and model comparison for three Nanog autoregulatory models, and describe possible extensions for future work.

The detailed characterization and analysis of Nanog time series performed in this thesis reveal that oscillations and excitatory behavior are unlikely to explain the observed heterogeneity, and that mESCs are capable of prolonged residence in a compartment with low Nanog expression with differing correlation structure and differentiation propensity to colonies containing mixed Nanog expression. Furthermore the tools presented provide new methods for investigating the dynamics of expression in mESCs and thus contribute to the growing body of knowledge of pluripotency regulation.

Acknowledgements

I would like to thank my colleagues Michael Strasser, Jan Hasenauer and Stefan Ganscha for fruitful discussions, and especially my thesis supervisors Carsten Marr and Fabian Theis for sustained guidance and feedback. Thanks also to Manfred Claassen for providing support and funding in the final stages of my Ph.D. I would also like to thank the European Research Council, the Deutsche Forschungsgemeinschaft, and the Helmholtz Zentrum München for funding opportunities.

And of course, my deepest gratitude to my wife Rounak, whose patience and perseverance always kept me on the right track.

Publications

1. **Feigelman, J.**, Theis, F. J., Marr, C. (2014). MCA: Multiresolution Correlation Analysis, a graphical tool for subpopulation identification in single-cell gene expression data. BMC Bioinformatics, 15(1), 1-10.

2. **Feigelman, J.**, Popović, N., Marr, C. (2015). A case study on the use of scale separation-based analytical propagators for parameter inference in models of stochastic gene regulation. Journal of Coupled Systems and Multiscale Dynamics, 3(2), 164173.

3. Filipczyk, A., Marr, C., Hastreiter, S., **Feigelman, J.**, Schwarzfischer, M., Hoppe, P. S., *et al.* (2015). Network plasticity of pluripotency transcription factors in embryonic stem cells. Nature Cell Biology, 17, 1235-1246.

4. Hilsenbeck, O., Schwarzfischer, M., Skylaki, S., Schauberger, B., Hoppe, P., Loeffler, D., Kokkaliaris, K., Hastreiter, S., Skylaki, E., Filipczyk, A., Strasser, M., Buggenthin, F., **Feigelman, J.**, Krumsiek, J., van den Berg, A., Endele, M., Etzrodt, M., Marr, C., Theis, F. J. A software platform for single-cell quantification of cellular and molecular behavior in long-term time-lapse microscopy. (submitted to Nature Biotechnology)

5. Strasser, M. K., **Feigelman, J.**, Theis, F. J., Marr, C. (2015). Inference of spatiotemporal effects on cellular state transitions from time-lapse microscopy. BMC Systems Biology, 9(1), 61.

6. Blasi, T., Feller, C., **Feigelman, J.**, Hasenauer, J. Imhof, A. Combinatorial Histone Acetylation Patterns Are Generated by Motif-Specific Reactions. Cell Systems 2, 4958 (2016).

Contents

I Background **xv**

1 Introduction **1**
1.1 Biological background . . . 2
1.1.1 Mouse embryonic stem cells . . . 2
1.1.2 Regulation of pluripotency . . . 2
1.1.3 Nanog heterogeneity . . . 4
1.1.4 mESC subpopulations and heterogeneity . . . 5
1.1.5 Experimental techniques . . . 6
1.1.6 Stochastic gene expression . . . 8
1.2 Models of stochastic gene expression . . . 9
1.3 Inference for gene regulation models . . . 12
1.4 Modeling of Nanog expression dynamics . . . 13
1.5 Overview of this thesis . . . 14

2 Methods **19**
2.1 Introduction . . . 19
2.2 Probability, statistics, and parameter inference . . . 20
2.2.1 Probability basics . . . 20
2.2.2 Bayes' rule . . . 22
2.2.3 Continuous random variables . . . 22
2.2.4 Statistical moments . . . 23
2.2.5 Hypothesis testing and p-values . . . 25
2.2.6 Parameter inference . . . 25
2.3 Dynamical systems . . . 30
2.3.1 Deterministic processes . . . 30
2.3.2 Stochastic processes . . . 33
2.4 Chemical physics . . . 37
2.4.1 Chemical reaction networks . . . 37
2.4.2 Chemical master equation . . . 40
2.4.3 Stochastic simulation . . . 47

II Results **49**

3 Multiresolution Correlation Analysis **51**

3.1 Introduction . . . 51
3.2 Results . . . 53
3.2.1 MCA reveals differential regulation of subpopulations in simulated gene expression data . . . 53
3.2.2 MCA plots as a diagnostic tool for transcriptomic analysis . . . 55
3.2.3 MCA provides additional insight into previously described subpopulations . . . 57
3.3 Discussion . . . 59
3.4 Conclusion . . . 60
3.5 Methods . . . 61
3.5.1 Estimation of correlations . . . 61
3.5.2 Multiresolution correlation analysis . . . 61
3.5.3 Construction of MCA plots . . . 62
3.5.4 Implementation . . . 62
3.5.5 Stochastic simulation . . . 63
3.5.6 Analysis of transcriptomic data . . . 64
3.6 Tables . . . 64

4 A case study on the use of scale separation-based analytical propagators for parameter inference in models of stochastic gene regulation 67
4.1 Introduction . . . 68
4.2 Methods . . . 69
4.2.1 Two-stage Gene Expression Model . . . 69
4.2.2 Propagator Expressions . . . 70
4.2.3 Special Cases of the Hypergeometric Functions . . . 72
4.2.4 Stochastic Simulation . . . 73
4.2.5 Implementation . . . 73
4.3 Results and Discussion . . . 73
4.3.1 Protein Time Courses Simulated With Gillespie's Algorithm . . . 74
4.3.2 Parameter Inference . . . 74
4.3.3 Comparison of Propagator Accuracy and Efficiency . . . 76
4.4 Conclusion . . . 80

5 Inferring gene regulation models using particle filtering 83
5.1 Introduction . . . 83
5.2 Mathematical background . . . 85
5.2.1 Bootstrap particle filter . . . 85
5.2.2 Gamma priors . . . 88
5.2.3 Model comparison . . . 89
5.3 Implementation . . . 90
5.4 Application to simulated data . . . 91
5.4.1 Models investigated . . . 91
5.4.2 Tree simulations . . . 92
5.4.3 Choice of prior and model parameters . . . 92
5.4.4 Parameter inference on single cells . . . 94

5.4.5 Parameter inference on genealogies . . . 100
5.5 Conclusions and outlook . . . 104

6 Analysis of NanogVENUS cellular lineages 109
6.1 Introduction . . . 109
6.2 Experimental setup . . . 110
6.2.1 Generation of mESC fluorescent fusion protein line . . . 110
6.2.2 NanogVENUS quantification . . . 110
6.3 Characterization of Nanog dynamics . . . 112
6.3.1 Oscillations . . . 112
6.3.2 Transitions . . . 118
6.3.3 Onsets . . . 121
6.3.4 Memory . . . 127
6.4 Identification of subpopulations . . . 132
6.5 Stochastic auto-regulatory models for NanogVENUS dynamics . . . 138

7 Discussion 145

Appendices 149

A Common probability distributions 151
A.1 Normal distribution . . . 151
A.2 Log-normal distribution . . . 151
A.3 Exponential distribution . . . 151
A.4 Poisson distribution . . . 152
A.5 Gamma distribution . . . 153
A.6 Gauss hypergeometric distribution . . . 153

B Gene regulation models 155
B.1 Birth-death model . . . 155
B.2 Two-stage model (mRNA and protein) . . . 156

Part I

Background

Chapter 1

Introduction

Mammalian life begins with the fertilization of an egg to give rise to the unicellular zygote which then divides repeatedly during embryonic development. The embryonic cells divide in a very regulated and predictable sequence of events, and as they divide the cells begin to take on specialized roles and to differ in their gene expression patterns leading to irreversible cell lineage decisions. Although the process is regular enough to be predicted with high accuracy, the fate of individual cells is nonetheless not entirely deterministic.

Embryonic stem cells (ESCs) are cells which are isolated from the early-stage developing embryo and which can be maintained indefinitely in culture in a suitable medium. ESCs are *pluripotent*, i.e. they may give rise to all tissues of the adult organism, and self-renewing. Thus ESCs are essential not only to normal embryonic function and development, but also may be of great clinical relevance. A detailed understanding of the biological regulation of ESCs and of pluripotency in general is of critical importance for the development of novel therapies targeted at repairing damaged or dysfunctional tissues, importantly for cancer, spinal injuries, and organ replacement.

Despite their importance, the mechanisms regulating pluripotency and differentiation are only poorly understood. Although hundreds of factors are known to play a role in the establishment and maintenance of pluripotency, the precise quantitative nature of the interaction between essential elements of this pluripotency network is unknown. However, designing rational stem cell-based therapies and protocols for directed differentiation requires a detailed, mechanistic model for ESC function.

In this thesis, I take steps in this direction by developing tools that enable a quantitative description of the dynamics of Nanog, a key molecule for the maintenance of pluripotency in ESCs. I use novel data derived from mouse ESCs (mESCs), a widely used model organism for studying embryonic development and stem cell function, in order to describe Nanog dynamics at the single cell level using a collection of deterministic and stochastic models.

Structure of the thesis

Part I of this thesis contains the Introduction and methods providing the mathematical foundations for the subsequent material presented in Results. Part II contains the results of three investigations conducted during the course of this thesis, relevant to the investigation

of stochastic models of gene expression at the single-cell level, presented in Chapters 3, and 4. Chapter 3 and 4 are presented with minor modifications to the respective manuscripts.

In Chapter 5 I present a Bayesian inference algorithm for identifying mechanistic models from tree-structured, noisy, discrete protein time series. In Chapter 6, I present a detailed investigation of NanogVENUS expression dynamics and apply the inference algorithm developed in Chapter 5. The thesis concludes with the Outlook in Chapter 7. Appendices are included containing details for common probability distributions, and a detailed discussion of a simple gene regulatory model.

For a detailed synopsis of the contributions of this thesis, see Section 1.5

1.1 Biological background

1.1.1 Mouse embryonic stem cells

Mouse ESCs (mESCs) are isolated on embryonic day 3.5 from the inner cell mass (ICM), a collection of cells present in the interior of the blastocyst. The ICM contains cells which later develop into the embryo, while the exterior cells of the blastocyst develop into the extra-embryonic tissues which envelop and support the embryo during growth. Evans *et al.* demonstrated that the cells they isolated from the mouse embryo are capable of self-renewal, i.e. sustained replication, and differentiation into all cell types of the adult organism, a capability known as pluripotency [1]. When injected into donor mice, mESCs form teratomas, tumors containing a variety of cellular lineages, which serves as a test for pluripotency [2].

Following isolation, mESCs are separated and grown on a layer of embryonic fibroblast feeder cells in a medium derived from fetal bovine serum (FBS) which contains growth factors promoting pluripotency [1, 3], see Figure 1.1. However, both feeder cells and FBS exhibit considerable heterogeneity, leading to potential inconsistencies. Williams *et al.* discovered that adding the cytokine leukemia inhibitory factor (LIF) improves the maintenance of pluripotency in mESCs and replaced the soluble factor differentiation inhibitor activity (DIA) in culture protocols [4]. Smith later showed that it is possible to culture mESCs in a medium containing FBS and LIF, while dispensing with feeder cells entirely [5]. Lastly, the factor bone morphogenetic protein 4 (BMP4) has been shown to suppress differentiation into the neural lineage [6], obviating the need for FBS and thus rendering mESC culture protocols more controllable and consistent. Hence it is possible to culture mESCs under feeder and serum-free conditions. Protocols both with and without feeder cells are in use currently, with choice of protocol a matter of taste, depending e.g. on the resources and expertise of the research institute; however, feeder-free conditions have been shown to be more effective at preventing mESC differentiation, see e.g. for comparison of mESC culture protocols [7].

1.1.2 Regulation of pluripotency

Hundreds of transcription factors and microRNAs are known to be involved in the maintenance of pluripotency, see e.g. the PluriNetWork, a database of 574 known interactions in mESCs for details [8]. In particular, pluripotency is regulated by a set of "master regulators" that together form a core network consisting of the genes Oct4, Klf4, Sox2 and

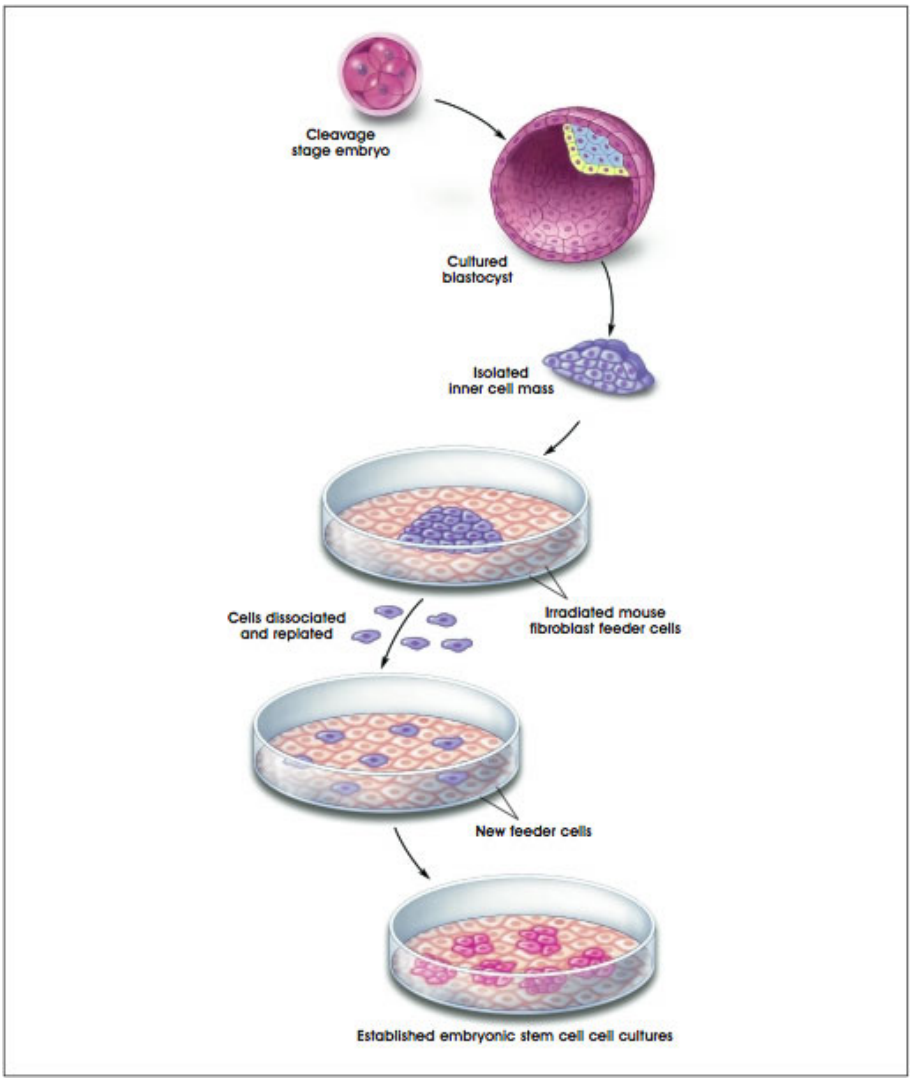

Figure 1.1: Embryonic stem cells are isolated from the inner cell mass of the blastocyst on embryonic day 3.5. After isolation, mESCs are grown in culture with feeder cells in a medium containing growth factors that inhibit differentiation. More recent protocols permit the culture of mESCs without feeder cells or serum, using the cytokines LIF and BMP4, which help prevent differentiation. Image copyright of Terese Winslow, reproduced with permission.

Nanog [9], as well as a host of peripheral regulators of differentiation such as Esrrb [10], Stat3, Tcf7, Sall4, LRH-1 and others [11, 12].

At the heart of the network is the factor Nanog, a homeodomain-containing transcription factor, essential for the maintenance of pluripotency. Nanog deficient cells differentiate into extraembryonic endodermal lineages [13], while Nanog over-expressing cells are capable of maintaining self-renewal in the absence of LIF/Stat3, suggesting a pathway for the maintenance of pluripotency distinct from other core pluripotency factors such as Oct4 and Sox2 which interact with the LIF/Stat3 pathway [14, 15]. Moreover, cells for which Nanog is low or absent possess a higher propensity for differentiation into the endodermal lineage, leading to the so-called 'ground state' hypothesis that Nanog is inherently highly expressed in mESCs, and only upon (transient) excursion into a low-expression state due to extra-cellular signaling or inherent stochasticity, are the cells at risk of differentiation [16].

Nanog binds to the promoters of many downstream lineage-determining genes (often together with Sox2 and Oct4) [17], and interacts physically with many proteins in the coordinated regulation of pluripotency in mESCs [18], see Saunders *et al.* [19] for a concise review of mESC regulation via Nanog. Nanog itself is regulated by a collection of pluripotency factors, including Oct4, Sox2, Klf4, Tcf3, Gcnf, Cdx2, Esrrb and many others [8]. Furthermore, Nanog is known to bind its own promoter [20], although the mechanism of autoregulation is yet unknown with evidence both for positive autoregulation [17, 21] and negative autoregulation [22, 23].

1.1.3 Nanog heterogeneity

Interestingly, although Nanog is a core regulator of pluripotency in mESCs, it is not uniformly expressed in mESC colonies as shown by Chambers *et al.* [24], see Figure 1.2. Chambers *et al.* further showed that flow-cytometry-sorted subpopulations with low Nanog expression were able to regenerate the unsorted distribution after several days of culture; the same is true for subpopulations sorted for high Nanog expression. Collectively, these results suggest that Nanog is heterogeneously expressed in mESCs, and that this heterogeneity arises via dynamic transitions through the range of possible expression levels.

Although Chambers *et al.* utilized a transcriptional reporter, the observed heterogeneity extends to the protein level. Indeed, recent studies have revealed that individual mESCs are capable of stochastically exploring the Nanog expression landscape, possibly potentiating the response of mESC colonies to extra-cellular differentiation cues [25, 26]. Meanwhile, Nanog low cells may be more prone to differentiation, and express markers for primitive endoderm [27].

The origin of Nanog heterogeneity is unclear. For instance Nanog heterogeneity can be a result of a hard-wired "epigenetic landscape" which gives rise to phenotypically-varying clonal subpopulations as a consequence of the underlying gene regulatory network [28]. Alternatively, Nanog heterogeneity could be due to periodic fluctuations in expression level due to a negative feedback loop [29]. Others have hypothesized that "chaotic oscillations" provide a mechanism whereby pluripotent cells randomly transit through meta-stable attractors corresponding to different cell lineages [30]. Moreover, mESCs can undergo epigenetic modifications such as histone acetylation and methylation which lead

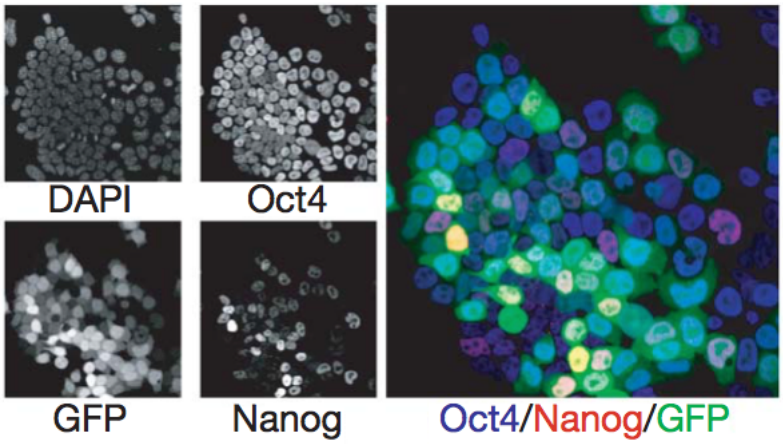

Figure 1.2: Nanog is heterogeneously expressed in mESCs. Immunohistological staining (left) of Oct4, Nanog and DAPI (a marker of chromatin), and expression of GFP under the control of the Nanog promoter reveal a subpopulation of cells for which Nanog/GFP is not expressed. This is in contrast to the mESC marker Oct4 which is highly homogeneously expressed. The color overlay (right) reveals many (blue) cells which are high in Oct4 but which do not express Nanog. Image taken from Chambers *et al.* [24].

to changes in promoter accessibility and overall expression levels of affected genes, giving rise to population-level heterogeneity [31–33]. Hence, Nanog heterogeneity emerges due to the complex interaction of multiple regulatory and epigenetic factors.

1.1.4 mESC subpopulations and heterogeneity

Heterogeneous expression is not unique to Nanog. On the contrary, it may be the case that Nanog is no more variable than other pluripotency genes [34]. Indeed heterogeneity of Nanog seems to be largely induced by the choice of culture medium and perhaps irrelevant to *in vivo* cellular decision-making [35]: culturing cells in a medium containing a two-inhibitor cocktail (2i), blocking glycogen synthase kinase 3 (GSK3) and mitogen-activated protein kinase (MAPK) seems to greatly decrease the propensity for differentiation, while upregulating the expression of pluripotency factors including Nanog [36]. Nanog heterogeneity is greatly reduced under 2i conditions, which has led to the hypothesis that mESCs exist primarily in a "ground state" that is perpetually self-renewing in the absence of extra-cellular differentiation cues mediated via the Stat3 pathway [16].

Other transcription factors have been previously shown to delineate undifferentiated mESC subpopulations, including endodermal lineage marker Hex [37], Gata6 [27], Rex1, Oct3/Oct4 [38], Stella, Pecam1, and SSEA1 [32], among others [39]. Moreover, mESC subpopulations potentially show qualitatively differing regulatory motifs evidenced by differential correlation networks [40]. Heterogeneous expression is hypothesized to afford plasticity to the cell in terms of subsequent fate determination—thus, the ultimate fate of the cell may be the consequence of a stochastic tug-of-war among competing lineage determining factors, orchestrated by extra-cellular signaling cues [41, 42].

This paradigm has given rise to a resurgence of interest in the hypothetical "epigenetic landscape", or Waddington's Landscape [43], in which cells "roll" down a potential energy surface, with local minima—or attractors—corresponding to cell lineage fates, see Figure 1.3. The pluripotent cell then rests upon a hill and is easily disturbed via internal stochasticity or external perturbation leading to the ultimate lineage commitment. This differentiation metaphor has been codified mathematically in a series of recent papers [44–50]; see e.g. Wu and Tzanakakis [51] for a review of mathematical and computational approaches for stem cell population heterogeneity.

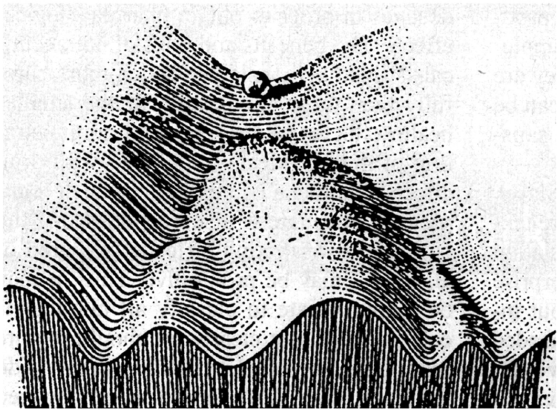

Figure 1.3: The epigenetic landscape posited by C.H. Waddington [43]. Cell lineage determination emerges by stochastic fate decision according to a predefined probability landscape.

1.1.5 Experimental techniques

The role of stochasticity in gene expression has been revealed in large part due to the advent of a slew of single-cell technologies, summarized in Figure 1.4. For example, fluorescent proteins including green fluorescent protein (GFP) and derivatives, make it possible to label individual molecules in single cells [52, 53]. Antibodies can be conjugated to fluorescent proteins and used to detect multiple surface proteins simultaneously using fluorescence activated flow cytometry (FACS). In FACS, the fluorescent probes are excited with a laser and the resultant emission is recorded for each cell individually. This is useful for both quantifying the surface protein abundance of individual cells (e.g. for determining cell type) and for sorting cells by their respective expression levels into various subpopulations. Analysis of the fluorescence intensity of various markers also makes it possible to identify unique populations with heterogeneous multidimensional expression profiles [54]. However, FACS is limited by the need for fluorescent probes with distinct emission spectra; significant spectral overlap leads to ambiguity among probes, and can lead to a false signal, see e.g. Oremord *et al.* for overview of FACS and probe design [55].

While FACS is thus limited to a small number of channels, another technology known as mass cytometry provides the means to quantify the abundances of many *intracellular* proteins with high accuracy. Mass cytometry works by conjugating rare metal ions to anti-

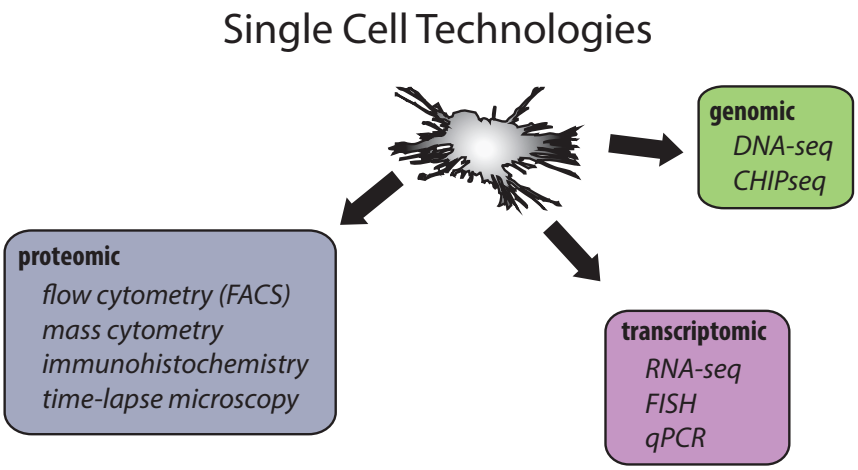

Figure 1.4: Single cell technologies allow investigation at the genomic, transcriptomic, and proteomic levels.

bodies against proteins of interest, followed by single-cell time-of-flight mass spectrometry. The rare metal ions can be quantified with high precision leading to an accurate estimate of the number of bound antibodies, and thus of the target protein abundance [56]. Mass cytometry hence provides a high-dimensional (currently up to about 100 proteins) snapshot of target protein abundances within single cells, and is useful for revealing cellular heterogeneity in tissues or clonal populations, especially in conjunction with likelihood-free clustering and dimensionality reduction methods such as tSNE [57], viSNE [58], SPADE [59], GPLVM [60], diffusion maps [61].

Although technologies such as FACS and mass cytometry provide quantitative protein information at the single-cell level, they do not permit the unique identification of each cell. That is, it is not possible to uniquely label the cells so as to maintain continuity from one measurement to the next. Moreover, in the case of mass cytometry, cells are destroyed during the measurement process making it impossible to gather longitudinal information about individual cells. Hence, both FACS and mass cytometry yield population *snapshots*, i.e. distributions at a fixed point in time. In contrast, time-lapse fluorescence microscopy (TLFM) facilitates longitudinal observations while maintaining information about cellular identity [62, 63]. Briefly, in TLFM one uses one or more lasers of different frequencies to excite the electrons of light-sensitive fluorophores within the cells. Fluorophores can be either endogenous, i.e. native to the cell, such as various metabolites, or exogenous such as quantum dots [64] or derivatives of GFP. The latter fluorophore is useful in conjunction with transgenic cell lines wherein a gene of interest is "knocked in", i.e. stably integrated into the genome with a functional promoter. Using this strategy one may derive transcriptional reporters where activation of the transgene promoter leads to the synthesis of a fluorescent protein, and translational reporters, where the functional gene product of the transgene is synthesized fused to a fluorescent protein thus maintaining a one-to-one ratio of fluorophores and transgene protein products. The laser-excited fluorophores spontaneously relax to more stable energy states generating photons at characteristic frequen-

cies. Using optical filters, it is possible to detect light of these wavelengths specifically and by quantifying the fluorescence intensity, it is possible to ascertain the total fluorophore concentration and thus that of the fused protein of interest as well.

Using TLFM, it is possible to gain information about processes with intrinsic heterogeneity within cellular populations, e.g. for the regulation of cell division [65], or cell cycle control [66]. TLFM has been successfully applied to reveal the underlying gene regulatory motif (such as positive autoregulation), in a system for which data generated from experiments using only bulk assays lead to the erroneous conclusion of no positive feedback [67]. TLFM can also be used to provide information about the spatial organization of proteins and transcripts, useful e.g. for understanding complex formation within single cells [68]. Moreover, the correlations among expression levels of fluorescently-labeled proteins within individual cells can be exploited to reveal information about the underlying regulatory mechanism, e.g. through the use of temporal cross-correlations [69, 70]. Thus, TLFM can be used to provide longitudinal information about mRNA and protein quantity and localization, which provides insight into intracellular regulation and organization. For an excellent review of single-cell TLFM, the reader is referred to the article by Muzzey and Oudenaarden [71].

Flow cytometry, mass cytometry and time-lapse fluorescence microscopy all provide information at the protein level. I note, however, that heterogeneity exists also at the transcriptional level as reflected by varying transcript counts of a particular gene throughout a cellular population. Transcriptomic heterogeneity can be analyzed via a suite of technologies enabling the quantification of the mRNA content of single cells. These technologies include single-cell quantitative reverse transcription polymerase chain reaction (qRT-PCR) which measures the relative abundance of a small number of genes targeted with gene-specific probes [72]; Fluorescence In Situ Hybridization (FISH), a biochemical method utilizing fluorescent probes which agglomerate with high-specificity to target mRNA molecules, rendering them visible as blobs which are detectable using standard fluorescence microscopy [73, 74]; and a collection of RNA sequencing (RNA-seq) methods for detecting single transcripts and associated polymorphisms without gene-specific probes. The latter can be combined with various along with various single single-cell setups such as microfluidic chips [75, 76] or droplet-based devices [77, 78] allowing very high throughput single cell transcriptomic profiling. Cellular transcriptomic contents can also be profiled using non-destructive techniques such as the MS2 system for detecting single transcripts as they are synthesized in living cells [79]. For further information on modern single-cell transcriptomic technologies, the reader is referred to Wang *et al.* and others [80–83].

1.1.6 Stochastic gene expression

Gene expression heterogeneity at the single-cell level is at least in part due to molecular stochasticity. Fundamentally, gene expression is a stochastic process modulated by the availability of transcription factors, polymerases, etc. involved in the transcriptional machinery [84]. Copy number variations in the components of the transcription process contribute to “extrinsic” variability, and affect the expression dynamics of all genes in the cell. In contrast, “intrinsic” stochasticity emerges due to the inherently probabilistic events, such as transcription factor binding events, that lead to eventual gene expression.

Contributions from intrinsic and extrinsic stochasticity, sometimes called "noise", can be quantified via the use of two-color dual knock-ins for a gene: covariance in the signal of the two reporters indicates synchronous fluctuations in gene activity, and is thus attributable to extrinsic stochasticity [85]. The residual variance in the expression of each gene is thus of intrinsic origin. However, initial investigations with two-color reporters were likely incorrect in their analysis of the intrinsic and extrinsic noise components, due to e.g. cell-cycle induced correlation of gene expression levels, thus care must be taken with this assay [86].

Noise seems to be a fundamental and essential phenomenon in cellular function [87]. It might for example help to coordinate regulation among large sets of genes, prime populations of cells for response to variegated stimuli, or potentiate evolution e.g. in bacterial colonies [88]. Noise may also facilitate "stochastic state switching", e.g. in development, stress response and cancer, and thus may be crucial for effectively targeting cancer therapies to tumors composed of heterogeneous subpopulations [89, 90].

Recent studies using FISH [74], which facilitates the quantification of individual mRNA molecules in single cells, have highlighted the bursty nature of gene expression: mRNA is not synthesized continuously, but rather in punctuated bursts. Bursty production of mRNA can give rise to highly variable distributions of proteins in cellular populations, which may in turn lead to differing phenotype. Hence, intrinsic noise plays a substantial role in the ultimate expression dynamics of individual cells, see e.g. Raj *et al.* for a review of the role of stochastic gene expression [84]. However, not all "variability" is due to stochastic effects: cells may respond predictably to various stimuli, environmental cues, spatial effects, etc. [91], all of which may generate non-stochastic variability which must be properly accounted for.

1.2 Models of stochastic gene expression

Due to the small numbers of DNA and mRNA molecules involved in the regulation of genes, it is typically necessary to devise models which explicitly account for stochasticity, see e.g. [92]. For example, deterministic models are insufficient to predict the dynamic behavior of distributions pertaining to lactose uptake in E. *coli* cells, a task which is made possible by the addition of a noise process, capturing inherent cellular variability [93]. Moreover, deterministic models of are typically qualitatively incorrect when blithely applied to small reaction volumes where stochastic effects due to discrete molecule numbers dominate, such as for gene expression at the single-cell level [94].

Gene expression is often modeled using chemical reaction networks (CRNs), a formalism which describes a collection of chemical species, the reactions that may take place between them, and their respective reaction rates which depend on the reaction stoichiometry and chemical kinetic constants [95], see Section 2.4.1. As for the stochastic gene regulation models, CRNs are described by a Markov jump process (MJP), and the evolution of the probability density describing the instantaneous configuration of the system in terms of molecular copy numbers obeys a (potentially infinite) differential-difference equation, known as the chemical master equation (CME) [96]. Solving the CME exactly is generally impossible except for very simple systems, and thus necessitates the use of numerical or analytical approximations. As an example, consider a CRN containing two

species X and Y. X is produced with a constant rate, reversibly dimerizes to form Y, and degrades at another rate:

$$\begin{aligned} \emptyset &\xrightarrow{k_0} X \\ X + X &\xrightarrow{k_1} Y \\ Y &\xrightarrow{k_2} X + X \\ X &\xrightarrow{k_3} \emptyset \end{aligned}$$

where the constants $k_0, \ldots$ indicate the associated reaction rates. The dynamics of this CRN can be approximated using ordinary differential equations where the concentration of each species evolves deterministically as

$$\begin{aligned} \frac{d}{dt}[X] &= k_0 + k_2[Y] - (2k_1 + k_3)[X] \\ \frac{d}{dt}[Y] &= k_1[X]^2 - k_2[Y]. \end{aligned}$$

Alternatively, the reaction could be modeled stochastically such that only the probability of a particular configuration of molecules of X and Y at time t is known. In the stochastic formulation, the probability of each reaction occurring within infinitessimal time dt depends on the present state of the system:

$$\begin{aligned} P(x+1, y, t+dt|x, y, t) &= k_0 \Omega dt \\ P(x-2, y+1, t+dt|x, y, t) &= k_1 x(x+1)/(2\Omega) dt \\ P(x+2, y-1, t+dt|x, y, t) &= k_2 y dt \\ P(x-1, y, t+dt|x, y, t) &= k_3 x dt \end{aligned}$$

where each expression is to be understood as the probability of a certain configuration at time $t + dt$ conditional on the configuration at time t; the term Ω denotes the reaction volume, which is important for the stochastic CRN description. Finally, by combining the above equations, Taylor expanding in the arguments x and y, and setting $dt \to 0$, one arrives at the CME formulation:

$$\begin{aligned} \frac{d}{dt}P(x, y, t) = {} & k_0 \Omega P(x-1, y, t) + k_1(x-1)x/(2\Omega) P(x+2, y-1, t) \\ & k_2(y+1)P(x-2, y+1, t) + k_3(x+1)P(x+1, y, t) \\ & - \left[k_0\Omega + k_1 x(x+1)/(2\Omega) + k_2 y + k_3 x\right] P(x, y, t) \end{aligned}$$

In principle the CME has to be solved for all states X, Y of interest. In this case the state space is potentially infinite since no constraints or conservation relationships exist.

Over the past few decades, a wide variety of stochastic models have been developed for describing gene regulation and biochemical reaction networks utilizing various analytical simplifications. Due to the discrete nature of molecules, the dynamics of stochastic biochemical reaction networks are typically modeled as a continuous-time, discrete-space

trajectory corresponding to a MJP, see Section 2.3.2. The time-dependent probability distribution of the system may be approximated many ways. For example, one may treat the MJP as a diffusion process with the Fokker-Planck approximation, a second order partial differential equation for the probability density, see Section 2.4.2. If one assumes a memoryless noise process, then one may approximate the stochastic system using the Chemical Langevin Equation, a stochastic differential equation, see Section 2.4.2; samples from this approximate process can be generated with numerical routines such as the Euler-Maruyama algorithm [97]. The system size expansion (see Section 2.4.2) expands the operator describing the evolution of the system's probability distribution in terms of a power series in the parameter describing the system's reaction volume, thus providing an approximation with asymptotically bounded truncation error. Moment equations capture the dynamics of statistical features of the true stochastic system, potentially reducing computational overhead [98]. Thus many mathematical and computational techniques exist for tackling stochasticity at the cellular level, see e.g. Wilkinson *et al.* [92] for a high-level review of stochastic models for chemical reaction networks.

Besides approximations for generic MJPs and CRNs, much analysis has been done for gene expression models in particular. For example, models have been developed which explicitly account for bursty transcription and/or translation and positive or negative autoregulation [99–108], see e.g. [109, 110] for an overview. However, such models typically only solve for the (approximate) steady-state distribution of the stochastic system since this is more tractable analytically than the full time-dependent solution. For some systems, analytical approximations are possible for the time-dependent dynamics, such as a system with constitutively active DNA, and mRNA and protein undergoing a birth-death process. If one assumes very fast degradation of mRNA relative to the degradation of protein, then the associated (approximate) probability distribution is computable [111], see Section B.1. Extensions to this method relax the assumption of infinite scale separation between mRNA and protein degradations, and thus lead to asymptotically more accurate expressions for the probability distributions [112, 113]; inference with such a system is investigated in Chapter 4. Furthermore, beyond intrinsic heterogeneity arising from the probabilistic evolution of the system according to the MJP, variability is also introduced e.g. by asymmetric cell division, which is difficult to distinguish from intrinsic noise [114–116].

Exact samples of trajectories of the process described by the CME—including the sequence of reaction firings and their respective times—can be generated by the Stochastic Simulation Algorithm (SSA) [117]. Many exact and approximate variations of the SSA exist which e.g. exploit algebraic tricks to expedite computation [118–121], approximate the Markov jump process using Poisson statistics [122–124], approximate discrete species by continuous variables [125, 126], exploit time-scale separation between fast and slow reactions (or species) [127–129], etc., in order to accelerate the SSA. For a thorough overview of approximate stochastic simulation algorithms, see e.g. [130].

Alternatively, instead of drawing samples from the (approximate) solution of the CME, one may directly solve it using numerical algorithms. If the state space is unlimited (i.e. the molecular copy number vector is unbounded) then solving the CME numerically is impossible. However, if one introduces an artificial upper bound, then the evolution of the CME on the reduced state space can be solved by simply integrating the resulting linear

equation [131]. Recent algorithms have been developed that exploit a factorization of the probability tensor of the system, which reduces storage requirements and the computation necessary for the direct solution of the CME [132]. Other approaches solve the CME not for the complete state space, but rather for a high probability density subregion, reducing computational effort [133, 134].

1.3 Inference for gene regulation models

The methods described in the previous section provide the means to either sample from the underlying stochastic process, compute the (approximate) time-dependent probability density of a fully-specified CRN, or derive analytical approximations to the steady state distribution of simple gene regulatory models. Thus each of these approaches is a means of solving the "forward problem", that is, predicting the (probabilistic) behavior of a system where the mechanisms and the associated model parameters are given.

However, one is typically more interested in the "inverse problem": inferring the correct model and parameters from partially observed, noisy data. If the (transient or steady-state) probability distribution can be computed, then the likelihood of the data for a particular parameter set can be evaluated, assuming a particular model structure. The problem then reduces to estimating the set of parameters for which this likelihood is greatest (the maximum likelihood estimator). Moreover, the likelihood function provides a mechanism by which to compare different parameter assumptions, i.e. parameter sets for which the likelihood does not greatly differ have roughly equal explanatory power, allowing one to generate confidence intervals for parameter estimates, see e.g. [135]. If prior information (codified as a probability distribution for model parameters) is included, one arrives at Bayesian inference, which provides the full "posterior" probability densities for model parameters—that is, the distributions after observed data are included [136]. Some variants of Bayesian inference, e.g. based on particle filtering approaches [137–140], also provide a framework for simultaneously inferring the trajectories of latent variables and parameters.

Parameter inference for stochastic systems is hard, particularly in the context of CRNs. It is complicated by the fact that the CME cannot generally be solved, thus, the system's probability distribution is generally unknown and must be approximated. For example, one can approximate the dynamics of the stochastic process using a diffusion equation, wherein one replaces the discrete variables by continuous ones undergoing a Langevin diffusion [141]. Or one can compute instead the statistical moments of the system [98, 142], which may necessitate approximating higher order moments in order to break the infinite hierarchy of dependencies among moments [143–145]. Inference can then be performed by attempting to match the statistical moments of the trajectories with those of the observed data. However, such an approach is limited in the domain of small molecular counts where moment equations cannot accurately describe the true probability distribution, leading sometimes even to nonphysical descriptions such as negative variances, etc. [145].

1.4 Modeling of Nanog expression dynamics

Nanog dynamics have been the subject of heavy investigation since early studies revealed both heterogeneous expression and a critical role of Nanog in the maintenance of pluripotency [24]. Many attempts have been made to elucidate the mechanism underlying Nanog heterogeneity and for transcription factor expression dynamics of early embryonic tissues in general. For instance, an early model by Chickarmane *et al.* attempted to capture the cellular decision-making process and emergence of trophectoderm and endoderm lineages in the early embryo [146], using a set of ordinary differential equations (ODEs) for relevant transcription factors. Glauche *et al.* suggested a generic model that would reproduce the observed bimodal Nanog expression distribution observed by Chambers *et al.* [24], either via stochastic transitions between two meta-stable attractors, or via oscillations induced by an unknown third species "X" [29]. Kalmar *et al.* proposed an excitatory model that would give rise to transient excursions into the Nanog "low" state [25]. Ochiai *et al.* proposed that observed Nanog transcription frequencies are consistent with a simple telegraph model where DNA can assume active and inactive states [147]. Herberg *et al.* propose an ODE model of Nanog dynamics with explicit regard to culture conditions (LIF/serum and 2i) [148].

Despite these efforts, no definitive model has emerged that is sufficient to explain the underlying mechanism giving rise to heterogeneous Nanog protein expression and dynamics. Many models, for example the 2006 model by Chickarmane *et al.* [149], include the three "core" pluripotency factors Oct4, Sox2 and Nanog in a mutual positive feedback configuration, and predict the dynamics by further assuming deterministic behavior and Hill functions for transactivation/inhibition. In contrast, the actual regulatory action of Nanog is controversial—for example Fidalgo *et al.* and Navarro *et al.* have both provided evidence that Nanog may in fact be involved in *negative* autoregulation [22, 23]. While the proposed model can give rise to bistability with suitably chosen parameters, with a state with a high pluripotency factor state corresponding to mESCs, and a state with low expression corresponding to differentiated cells, the model is never fit to real data and thus is purely hypothetical. Moreover, the model does not account for Nanog expression heterogeneity, specifically, for ES cells which are high for pluripotency factors but low for Nanog expression. A later model by Chickarmane *et al.* extends [149] by including Cdx2 and Gata-6 which serve as proxies for the trophectoderm, and endoderm lineages, respectively [146], and Gcnf, a downstream target of both Cdx2 and Gata-6, which inhibits Oct4 expression. The model thus contains a negative feedback loop leading to down-regulation of pluripotency factors in conjunction with the expression of differentiation markers. However, the model again makes no attempt to fit real Nanog data, oversimplifies by assuming deterministic ODE dynamics for each factor, fails to explain Nanog heterogeneity, and does not account for potential subpopulations, thus providing at most qualitative insight into potential regulation mechanisms in cell fate determination.

Glauche *et al.* present two potential mechanisms tailored for generating bimodal distributions similar to that observed for Nanog transcripts: a stochastically bistable system and an oscillator involving negative feedback via some unidentified additional factor [29]. In principle both mechanisms are capable of generating a bimodal distribution compatible with observed Nanog heterogeneity. However, both models are obviously highly ideal-

ized. The stochastically bistable system gives rise to dynamics that are clearly unrealistic compared to real time series: the model generates dynamics with very separate high and low states (in real data the division is much less clear), and too rapid of transitions between compartments (compare with e.g. [26, 150]). The model parameters were tuned to achieve a stochastically bistable system, and thus it may be possible to adjust them to more closely agree with observed Nanog protein dynamics. Unfortunately, however, no published system currently exists which expresses reporters for all three markers simultaneously, making fitting such a model to real data very difficult. Furthermore, the model makes an *ad hoc* assumption about the mechanistic regulation of the pluripotency factors, whereas in principle one would like to infer the correct model in a more principled way, e.g. using model comparison, for example. Lastly, mRNA dynamics are completely neglected which could be a fatal shortcoming if e.g. post-transcriptional control proves to be an important regulatory mechanism for mESCs, see e.g. [151].

Several other models have been proposed for Nanog expression dynamics. Wu and Tzanakakis proposed a stochastic model incorporating random partitioning and random monoallelic expression which led to heterogeneous expression dynamics [152]. However, the assumption of monoallelism is based largely on a study by Miyanari *et al.* [153], which is contradicted by two later studies [34, 154], raising doubt as to the validity of such a model. Singer *et al.* surmise that Nanog protein heterogeneity may arise in part due to switching between states with differing translation rates [31]. This is consistent with the "telegraph" model espoused by Ochiai *et al.* , wherein they claim that observed Nanog transcription time series (as quantified via a fluorescent reporter) are well explained by a simple fluctuation between active and inactive chromatin configurations without requiring regulatory control via feedback [147]. The "stochastically bistable" model of Nanog dynamics was further characterized by Zhang *et al.* who derived equations of motion for Nanog expression from its probability landscape [155]. Other quantitative Nanog models were proposed by Herberg *et al.* [29, 148], and summarized in a recent review by the same author [156].

Thus, while many stochastic and deterministic models of Nanog regulation have been proposed in recent years, no definitive, quantitative model of Nanog regulation has emerged. In particular, no group has yet attempted to fit real Nanog protein time series using a stochastic model encompassing DNA, mRNA and protein, nor has a rigorous model comparison been performed. Rather, previous attempts have assumed simple models based on presumed topologies, and shown that such a model is hypothetically compatible with heterogeneous Nanog expression, if properly tuned via the model parameters. Thus it is of great interest to continue investigating the long-term dynamics of Nanog expression, with single cell resolution, with the aim of achieving a detailed description of observed Nanog dynamics, and potentially repudiating proposed models.

1.5 Overview of this thesis

In this thesis, I address the problems presented in the previous sections, which can be summarized as:

- Nanog heterogeneity is poorly understood. In particular it is unclear:

  - whether Nanog expression in ESC colonies is generated by different subpopulations with differing expression levels
  - how to identify subpopulations with different qualitative behaviors
  - how Nanog undergoes transitions between low and high states
  - the stability of such states, and
  - the underlying mechanism driving these transitions

- Quantitative, data-driven models of Nanog are lacking in the literature, due to:
  - lack of long-term time-lapse fluorescence microscopy data
  - difficulty in deriving approximate models for Nanog behavior at the appropriate scale
  - lack of methods for fitting stochastic models to tree-structured data

This thesis is concerned with the development of quantitative methods for providing insight into the behavior of mESCs, particularly in terms of the expression and regulation of Nanog, a key pluripotency regulator. However, the methods developed are applicable to any single-cell expression data. While much investigation has been performed on the mESC regulatory network, relatively little is known about the mechanistic regulation of Nanog, a keep regulator of pluripotency. Indeed it has yet to be shown definitively if Nanog undergoes positive or negative autoregulation. The majority of models presented are purely theoretical, with no rigorous attempt made to fit fully stochastic models, addressing DNA, mRNA and protein dynamics, to single cells or colonies of cells. Furthermore, no investigation has attempted to identify the correct regulatory model via model selection.

I present a detailed investigation into the dynamical behavior of individual mESCs, as characterized by their expression of a fluorescent Nanog protein reporter, NanogVENUS. In particular, I provide evidence that suggests that previously hypothesized models underlying the observed Nanog heterogeneity are incorrect. I specifically investigate oscillations, transitions between mother and daughter cells, memory/correlation between related cells, the behavior of other pluripotency factors in sister cells, onsets of NanogVENUS expression in low-sorted mESC populations, and the existence of subpopulations of mESCs which differ in their expression levels of pluripotency factors and in the correlation network of pluripotency factors. These analyses are largely contained in the manuscript (for which I am second author):

> Filipzyck, A., Marr, C., Hastreiter, S., **Feigelman, J.**, Schwarzfischer, M., Hoppe, P. S., et al. (2015). Network plasticity of pluripotency transcription factors in embryonic stem cells. Nature Cell Biology. http://doi.org/10.1038/ncb3237.

As a part of the aforementioned analysis, I developed a graphical analysis technique, Multiresolution Correlation Analysis (MCA), useful for the investigation of subpopulations in low-dimensional gene expression data, e.g. from qPCR or from fluorescence microscopy. This method is applied to the mESC data for investigating the existence of unique subpopulations with different partial correlation structures. I further applied MCA to published mESC qPCR data, as described in the published manuscript:

> **Feigelman, J.**, Theis, F. J., & Marr, C. (2014). MCA: Multiresolution Correlation Analysis, a graphical tool for subpopulation identification in single-cell gene expression data. BMC Bioinformatics, 15(1), 1-10.

While investigating Nanog regulation, I attempted to utilize a new analytical technique employing a geometric singular perturbation approach applied to a simple two-stage model of gene expression. The two-stage model is a commonly used approximation of gene expression, and the method [112] provides an analytical approximation to the transition density probability distribution of the stochastic system in terms of proteins and mRNAs. The method had not previously been applied to the problem of parameter inference, thus I investigated its utility in a case study, comparing it against the previous zeroth-order model which assumes infinite scale separation between mRNA and protein degradation. The results are published in the following manuscript, and reveal that the new method is surprisingly not appropriate for parameter inference due to the frequent occurrence of negative transition densities arising from the approximation used:

> **Feigelman, J.**, Popović, N., Marr, C. (2015). A case study on the use of scale separation-based analytical propagators for parameter inference in models of stochastic gene regulation. Journal of Coupled Systems and Multiscale Dynamics, 3(2), 164173. http://doi.org/10.1166/jcsmd.2015.1074

Lastly, I implemented a Bayesian, fully stochastic single-cell inference algorithm capturing DNA, mRNA and protein dynamics, based on the bootstrap particle filter, see Chapter 5. I present the mathematical underpinnings of the algorithm, and applications to simulated data generated from three regulatory models: no regulation, and positive or negative transcriptional autoregulation. The algorithm presented is the first method to directly perform fully exact, Bayesian inference on noisy, partially- and discretely-observed, *tree-structured* protein time series. Using the synthetic data, I show that the method provides a better estimate of model parameters than possible without the additional modeling of the cell division process, and substantially improves the accuracy of the model identification over single-cell based methods. Finally, I applied the inference algorithm to real data obtained from a fluorescent Nanog protein reporter mESC line and report the findings of model selection using Bayes factor analysis and the particle filtering inference algorithm.

I have also contributed to the following manuscripts which are not discussed in this thesis:

- Michael Schwarzfischer, Oliver Hilsenbeck, Bernhard Schauberger, Sabine Hug, Adam Filipczyk, Philipp S. Hoppe, Michael Strasser, Felix Buggenthin, **Justin Feigelman,** Jan Krumsiek, Dirk Löffler, Konstantinos D. Kokkaliaris,, Adrianus J. J. van den Berg,, Max Endele, Jan Hasenauer, Carsten Marr, Fabian J. Theis, Timm Schroeder. Reliable long-term single-cell tracking and quantification of cellular and molecular behavior in time-lapse microscopy (in preparation)

- Strasser, M. K., **Feigelman, J.**, Theis, F. J., Marr, C. (2015). Inference of spatiotemporal effects on cellular state transitions from time-lapse microscopy. BMC Systems Biology, 9(1), 61. http://doi.org/10.1186/s12918-015-0208-5

- Thomas Blasi, Christian Feller, **Justin Feigelman**, Jan Hasenauer, Axel Imhof, Fabian J. Theis, Peter B. Becker, Carsten Marr. Combinatorial histone acetylation patterns are generated by motif-specific reactions. (submitted)

Chapter 2

Methods

2.1 Introduction

In this chapter I present the tools necessary for understanding the results presented in later chapters. In particular, the results largely derive from the application of probability theory to identify models and sets of model parameters that provide the best fit to observed data. Thus, I present basic results from probability theory necessary for understanding the inference techniques used in the study of Nanog time series. I then present results from dynamical system theory which are necessary for understanding the deterministic and stochastic models derived. Lastly, I rederive results from chemical kinetic theory to provide the theoretical basis for chemical reaction networks and their probabilistic description via the chemical master equation.

Modeling

Modeling is typically used to gain insight into the mechanism underlying observed data and to predict the results of future experiments. The same modeling strategies can be applied to data from very diverse sources, spanning from turbulence in jet engines to cellular growth and gene expression. A wide range of mathematical and computational techniques are used for modeling experimental data. At the core however, the essential goal remains the same, namely to devise a system to predict the value of some observable variables which can then be compared with experimental results. Hence, a good model is one that agrees with experimental results, does not contradict known facts, and has reasonable core assumptions. Moreover, a model should be falsifiable in the sense that it should be possible to validate the model by making predictions, for example by varying model parameters, and that these predictions can be compared with the results of additional experiments.

A model consists of variables corresponding to experimental covariates, and a set of deterministic or probabilistic rules describing the relationship among these variables. Models can be static or dynamic, depending on whether the variables are time-dependent: static models describe the relationship among covariates whereas a dynamical model describes the deterministic or probabilistic evolution of the system. Whereas a model is partly defined by the statistical or dynamical relationship amongst covariates, it is also described by the set of model parameters, i.e. a set of variables in addition to the covariates of the

model, necessary for the evaluation of the state of the system. The performance of a given model and parameter choice can be assessed on the basis of the power of model parameters to explain the observed data, requiring a measure of goodness of fit. This measure can be any arbitrary score related to the discrepancy between model prediction and actual observation. A common choice is the sum of the square of residual errors, i.e. the difference between the observed data and the model prediction. More sophisticated methods rely on a measure of the probability of observing a particular outcome or observation based on the assumed model and parameters; good models will produce outcomes that resembled the observed data with a high probability.

2.2 Probability, statistics, and parameter inference

2.2.1 Probability basics

In this section I provide a very brief summary of basic probability. For a more thorough treatment, see for any textbook on elementary probability theory, e.g. [157]. It is assumed that the reader is familiar with sets and basic set operations such as unions, intersections and set differences.

We define a **state space** Ω to be some set of possible outcomes of a random event. For example when flipping of a coin, the possible outcomes are heads or tails, denoted H and T, respectively. Thus the state space for this system is $\Omega = \{H, T\}$.

An **elementary event** $\omega \in \Omega$, is a single element of the state space. For the coin, the elementary events are H and T. More generally, an **event** is some subset of the state space that satisfies a given condition. For example, when tossing a die with outcome $\omega \in \Omega = \{1, 2, 3, 4, 5, 6\}$, an event A might be defined as $A = \{\omega \geq 3\} = \{3, 4, 5, 6\}$.

An event is **measurable** if it is possible to determine from an event $\omega \in \Omega$ whether $\omega \in A$. Events can be combined via and/or relationships. The compound event consisting of event A **or** event B, $(A, B \in \Omega)$ is given by the union $A \cup B$. The event A **and** B is given by the intersection, $A \cap B$. The **complement** of an event is the event which corresponds to the "opposite" of an event. For example, for the dice if $A = \{X \geq 3\}$, then the complement $A^c = \{X < 3\}$, and in general is given by the set difference $A^c = \Omega \setminus A$.

A **σ-algebra** $\mathcal{A}$ of Ω is a subset of 2^Ω, the set of all subsets of Ω, with the following properties:

- The empty set $\emptyset \in \mathcal{A}$
- The state space $\Omega \in \mathcal{A}$
- If an event $A \in \mathcal{A}$ then the complement of the event $A^c \in \mathcal{A}$
- The set $\mathcal{A}$ is closed under countable set operations

The state space Ω and the σ-algebra $\mathcal{A}$ together constitute a **measurable space** $(\Omega, \mathcal{A}, P)$. The probability measure P gives the "size" of each element of the σ-algebra, a number between 0 and 1, with $P(\Omega) = 1$ and $P(\emptyset) = 0$; the events $\emptyset$ and Ω are sometimes referred to as the *impossible* event and *certain* event, respectively. For a countable state space Ω, the probability measure assigns to each $\omega \in \mathcal{A}$ a number $P_\omega \in [0, 1]$. Moreover, the

probability of any union of disjoint elements $X, Y \in \mathcal{A}$, i.e. $X \cap Y = \emptyset$, is given by the sum of the two probabilities: $P(X \cup Y) = P(X) + P(Y)$. More generally $P(\cup_{n=1}^{\infty} A_n) = \sum_{n=1}^{\infty} A_n$ for a series A_n of pairwise disjoint elements of $\mathcal{A}$. If X and Y are not disjoint, the probability is given by $P(X \cup Y) = P(X) + P(Y) - P(X \cap Y)$.

A **random variable** (RV) is a function of an outcome $\omega \in \Omega$, therefore its value is determined by the outcome of the event. The probability of a RV $X(\omega)$ assuming a particular value x is the summed probability of all events yielding that value: $P(X = x) = \sum_{\omega \in \Omega} P(\omega) \mathbb{1}_{X(\omega)=x}$, where $\mathbb{1}$ is the indicator function.

If the outcome is not countable, e.g. $\Omega \subset \mathbb{R}$, then the probability $P(X = x)$ is obtained by integration: $P(X = x) = \int_{\Omega} \delta(X(\omega) - x) \mu(d\omega)$, where μ is the measure of the set ω, and $\delta(X)$ (the Dirac delta function) is one if $X = 0$ and zero otherwise.

Joint and marginal probability

The **joint probability** of a two disjoint events A and B ($A, B \in \Omega$) is the probability of the intersection of the two events: $P(A, B) = P(A \cap B)$. For example when tossing a die, the probability that the random variable $X(\omega) = \omega$ is both even-valued and larger than 3, is given by the probability $P(X \text{ even and } X > 3) = P(\{X \text{ even}\} \cap \{X > 3\}) = P(\{4, 6\})$. If X_1, X_2 are two random variables of the random outcome, i.e. $X_{1,2} : \Omega \to T$ (for some space T), then the joint probability of X_1, X_2 is given by $P(X_1 = x_1, X_2 = x_2) = \sum_{\omega} P(\omega) \mathbb{1}_{\{X_1(\omega)=x_1, X_2(\omega)=x_2\}}$, and similarly for multiple random variables.

The **marginal probability** of the RV $X_1 = x_1$ is the summed probability all events $\omega \in \Omega$ for which $X_1(\omega) = x_1$. The marginal probability can be obtained from the joint probability via summation over the other random variables of the joint probability, e.g. $P(X_1) = \sum_{x_2 \in T} P(X_1 = x_1, X_2 = x_2)$, where T is the range of the RV X_2. For instance, if X_1 is the result of one coin flip, and $X_2, X_3, \ldots$ the results of subsequent coin flips, then the probability $P(X_1 = H)$ is computed by summing the joint probabilities of all events for which the first coin toss resulted in heads, regardless of the outcomes of the remaining coin tosses:

$$\begin{aligned} P(X_1 = H) &= \sum_{\omega \in \Omega} P(\omega) \mathbb{1}_{\{X_1 = H\}} \\ &= \sum_{x_2, \ldots, x_N} P(X_1 = H, X_2 = x_2, \ldots, X_N = x_N) \end{aligned}$$

Conditional probability and conditional independence

A pair of random variables X, Y are **conditionally independent** if the joint density of X and Y satisfies $P(X = x, Y = y) = P(X = x)P(Y = Y)$, i.e. the joint probability is the product of the two marginal probabilities. If Y is not conditionally independent of X, then the probability of Y will change depending on the value of X. The **conditional probability** of Y given the value of X, or *conditional* on X, is denoted $P(Y|X)$. Thus conditional independence implies $P(Y|X) = P(Y)$. Conversely, if Y is not conditionally independent of X, then it is conditionally *dependent*. For example, if X and Y are the outcomes of two consecutive coin flips, then Y is conditionally independent of X, and the probability of the outcome $X = H$ AND $Y = H$ is given by $P(X = H) \times P(Y = H)$.

Factorization

The joint probability can be factorized as :

$$P(X,Y) = P(X|Y)P(Y) = P(Y|X)P(Y) \tag{2.1}$$

which can be intuitively interpreted as the probability that both X and Y occur is the probability that Y occurs *and* that X occurs, given that Y has occurred, or vice-versa.

2.2.2 Bayes' rule

From the factorization of the joint density (2.1) it follows that

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}. \tag{2.2}$$

This simple formula is known as **Bayes' Law**, and forms the basis of Bayesian statistics. In the context of parameter inference, it is commonly written as $P(\Theta|D) = \frac{P(D|\Theta)P(\Theta)}{P(D)}$, where Θ are the model parameters and D the observed data. $P(\Theta|D)$ is known as the **posterior probability** of the model, i.e. the probability of the model parameters after considering the data; the model parameters can become more or less probable depending on the agreement between model prediction and observed data. The term $P(D|\Theta)$ is known as the **likelihood** of Θ, i.e. the probability that the data D would be observed if Θ were true. The term $P(\Theta)$ indicates the probability of the model parameters in the absence of observed data, and is known as the **prior probability**. $P(D)$ is the marginal likelihood of the data, also known as the **evidence**, obtained by summing the probability of the data for all possible values of the model parameters, weighted by the prior probability of those model parameters.

The term $P(\Theta)$ plays a critical role in Bayesian statistics, since it represents prior knowledge about the probability of model parameters, for example from literature, experimental results, or physical feasibility. The prior reflects our knowledge of the true values of the parameters before the inclusion of the data D. Hence, the posterior likelihood $P(\Theta|D)$ represents out knowledge about the parameter values after including the data D.

2.2.3 Continuous random variables

In the case of a continuous RV $X : \Omega \to T$, we define the **probability density function** (PDF) of X, denoted $\phi(X)$ as the probability per unit volume (in the range space T) of X. That is, for a region $A \subseteq T$, $P(X \in A) = \int_A \phi(x)dx = \int_\Omega \delta(X(\omega) \in A)\mu(d\omega)$. For example, if the range of X is $T \subset \mathbb{R}$, then

$$P(X \in [a,b]) = \int_a^b \phi(x)dx \tag{2.3}$$

Unlike for discrete variables, the PDF need not be bounded above by one. By convention $\phi(\infty) = 0$, and $\phi(x) = 0, \forall x \notin T$.

Probabilities of events involving multiple continuous variables can be computed analogously to (2.3) by utilizing higher-dimensional integrals.

The **cumulative probability density** of a univariate random variable $X : \Omega \to T \subset \mathbb{R}$ is defined as:

$$\Phi(x) = \int_{-\infty}^{x} \phi(s)ds. \tag{2.4}$$

Using (2.4), one sees that $P(X \in [a, b]) = \int_a^b \phi(s)ds = \Phi(b) - \Phi(a)$.

2.2.4 Statistical moments

One is frequently interested in knowing the likely outcome of a random variable. For a discrete RV, one can compute the most likely outcome, i.e. the outcome for which $P(X)$ is maximal. For continuous random variables, the peak (or mode) of the probability density provides some indication of a likely outcome, however, one must also take into consideration the width of the peak, i.e. the relative uncertainty in the outcome. In particular, the **expectation** of a RV X with PDF $\phi(x)$, is defined as:

$$\mathbb{E}[X] = \int x\,\phi(x)dx \tag{2.5}$$

where the integral extends over the the range of X. For symmetric, unimodal probability distributions, the expectation coincides with the peak of the distribution. The expectation of a function $f(X)$ is defined analogously as $\mathbb{E}[f(X)] = \int f(x)\phi(x)dx$. If X is discrete-valued, then the expectation is computed using the probability mass function: $\mathbb{E}[X] = \sum_x xP(x)$.

More generally one may compute any **statistical moment** of the (univariate) distribution described by $\phi(x)$. The k^{th} statistical moment ($k = 0, 1, \ldots$) is defined as:

$$M_k = \mathbb{E}[X^k] = \int x^k \phi(x)dx \tag{2.6}$$

where $M_0 = \int \phi(x)dx = 1$ and the expectation is given by $\mathbb{E}[X] = M_1$.

The **variance** of X is defined as $\text{var}[X] = \mathbb{E}[(X - \mu)^2] = \mathbb{E}[X^2] - \mu^2$, where $\mu = M_1$. The variance provides an indication of the spread of the probability density function and relates to the variability or uncertainty of the random variable. The **standard deviation** is defined as $\sigma = \sqrt{\text{var}[X]}$.

Multivariate moments

If $X = (X_1, X_2, \ldots, X_N)$ is a vector of random variables, i.e. a multivariate RV, then probability density of X is given by the joint probability $P(X_1, X_2, \ldots, X_N)$.

Analogously to the moment equation (2.6), moments $M_{k_1,\ldots,k_N}$ can be defined for the multivariate RV as:

$$M_{k_1,\ldots,k_N} = \mathbb{E}[\prod_{i=1}^{N}(X_i - E[X_i])^{k_i}]. \tag{2.7}$$

The **order** of the moment $M_{k_1,\ldots,k_N}$ is given simply by $\sum_{i=1}^{N} k_i$.

Covariance

The **covariance** of a pair of RVs, or of a multivariate RV, is defined as $\text{cov}(X,Y) = \mathbb{E}[(X-\mu_X)(Y-\mu_Y)]$, where μ_X and μ_Y are the expectations of X and Y, respectively. If X and Y are conditionally independent, the covariance vanishes:

$$\begin{aligned}\mathbb{E}[(X-\mu_X)(Y-\mu_Y)] &= \iint (x-\mu_X)(y-\mu_Y)\phi(x,y)dxdy \\ &= \iint (x-\mu_X)(y-\mu_Y)\phi_x(x)\phi_y(y)dxdy \\ &= \int (x-\mu_X)\phi_x(x)dx \int (y-\mu_Y)\phi_y(y)dy \\ &= (\mu_X-\mu_X)(\mu_Y-\mu_Y) = 0\end{aligned} \tag{2.8}$$

where $\phi_x(x)$ and $\phi_y(y)$ are the densities of X and Y, respectively. The same holds for discrete RV (i.e. Ω is countable), for which $\mathbb{E}[X] = \sum_x xP(x)$, and $\text{cov}(X,Y) = \sum_x \sum_y xyP(x,y) - \mathbb{E}[X]\mathbb{E}[Y]$.

The covariance between two variables is a measure of the interdependence between the two variables, and zero covariance indicates no (linear) relationship between the variables, as shown in (2.8). The **covariance matrix** $\Sigma = \{\sigma_{ij}\}_{i,j=1}^N$ of a set of random variables has entries $\sigma_{ij} = \text{cov}(X_i, X_j)$, and $\sigma_{ii} = \text{var}(X_i)$.

Correlation and partial correlation

The **correlation** between two random variables X, Y is equal to the covariance scaled by the respective univariate standard deviations, σ_X and σ_Y:

$$\text{cor}(X,Y) = \frac{\text{cov}(X,Y)}{\sqrt{\text{var}[X]\,\text{var}[Y]}} = \frac{\text{cov}(X,Y)}{\sigma_X \sigma_Y} \tag{2.9}$$

While the range of the covariance $\text{cov}(X,Y)$ is $(-\infty, \infty)$, the correlation is bounded by -1 and 1, with -1 indicating complete *anti-correlation*, i.e. a negative, linear dependence of X and Y. For example, if $Y = -\lambda X$ for some $\lambda \in \mathbb{R}^+$, then

$$\begin{aligned}\text{cov}(X,Y) &= \mathbb{E}[(X-\mu_X)(-\lambda X + \lambda\mu_X)] \\ &= -\lambda\mathbb{E}[(X-\mu_X)(X-\mu_X)] = -\lambda \,\text{var}\, X \\ \text{var}\, Y &= \mathbb{E}[(-\lambda X + \lambda\mu_X)^2] = \lambda^2 \,\text{var}\, X \\ \text{cor}(X,Y) &= \frac{\text{cov}(X,Y)}{\sqrt{\text{var}\, X \,\text{var}\, Y}} \\ &= \frac{-\lambda \,\text{var}\, X}{\sqrt{\lambda^2 \,\text{var}\, X \,\text{var}\, X}} = -1\end{aligned} \tag{2.10}$$

hence Y is perfectly anti-correlated with X; if $Y = \lambda X, \lambda \in \mathbb{R}^+$ then $\text{cor}(X,Y) = 1$.

Eq. (2.9) defines the so-called **Pearson correlation**. However, correlation can be defined in other ways using other statistical measures such as *Spearman* or *Kendall* correlation, each having to do with the extent to which value of one variable depends on the value of another [158]. However, it is easy to see that correlation suffers transitivity, i.e.

random variables that are not directly interrelated may nonetheless show correlation due to indirect associations.

For example, consider a system of three linearly-related continuous random variables, X, Y and Z, such that $X \propto Y$ and $Y \propto Z$. Obviously $X \propto Y$, and thus $\text{cor}(X, Z) = 1$.

Thus, although X does not *directly* depend on Z in this trivial example, it nonetheless shows perfect correlation due to the dependence on the intermediate Y. However, if one conditions on the value of Y, then X and Z are no longer correlated.

More generally, one may compute the partial correlation between any two variables $X_i, X_j \in V$ for some system $V = \{X_1, \ldots, X_N\}$ of random variables as:

$$\rho_{ij|V\setminus\{X_i,X_j\}} = -\frac{p_{ij}}{\sqrt{p_{ii}p_{jj}}} \tag{2.11}$$

where $\rho_{ij|V\setminus\{X_i,X_j\}}$ is the **partial correlation** of X_i and X_j conditioned on the remaining variables in the set V [159]. The term p_{ij} indicates the $(i,j)^{th}$ entry in the matrix $P = R^{-1}$ where R is the correlation matrix defined as in (2.9). Alternatively, the partial correlation may be understood as the residual correlation between two variables X_i, X_j, after subtracting a linear regression upon the remaining variables $V \setminus \{X_i, X_j\}$; if the residuals correlate, it indicates the presence of a correlation component that is not explained by the other variables in V.

2.2.5 Hypothesis testing and p-values

In inference, it is frequently necessary to decide whether a given model or hypothesis is consistent with the observed data. One may define a **null hypothesis**, H_0, representing a certain model assumption, e.g. that the mean μ of the observed data is equal to a value μ_0, and a mutually-exclusive **alternative hypothesis**, H_A, such as $\mu \neq \mu_0$ or $\mu > \mu_0$. Assuming that the probability of the observed data (assuming the H_0 to be correct) can be computed, the null hypothesis is rejected if the probability of the observed data is less than a particular, small value, i.e. $P(D|H_0) < \alpha$ for some small but arbitrary α. The value α is known as the **significance**, and the probability $P(D|H_0)$ is known as the **p-value**, p. Typically, α is set to 0.05 for a "significant" result, and 0.01 or 0.001 for a "very significant" result, indicating a low probability of the data being observed if the null hypothesis were true. Thus, if $p < \alpha$ one can reject the null hypothesis in favor of the alternative hypothesis.

2.2.6 Parameter inference

In addition to identifying a model which is compatible with the observed data, one may also attempt to learn, or *infer* the set of model parameters Θ which are compatible with the data. The inference procedure may yield a single best estimate, or *point estimate*, for the parameters, or, as is often the case there may exist many possible sets of parameter values which could possibly give rise to the data, in which case the aim is to infer all possible parameters compatible with the data and their respective probabilities. For a given set of model parameters $\boldsymbol{\theta}$, and set of (multivariate) observations $\mathcal{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$, the likelihood of $\boldsymbol{\theta}$, $L(\boldsymbol{\theta})$, is given by the probability of observing the data, assuming those parameters: $L(\boldsymbol{\theta}) = P(\mathcal{X}|\boldsymbol{\theta})$.

If the residual errors are entirely due to the measurement process, then the errors are conditionally independent from one another and the likelihood factorizes as $L(\boldsymbol{\theta}) = P(\mathcal{X}|\boldsymbol{\theta}) = \prod_{i=1}^{N} P(\mathbf{X}_i|\boldsymbol{\theta})$. Furthermore, if one assumes normal measurement errors with variance σ^2, then the likelihood function is the product of the normal densities (see Appendix A.1):

$$\begin{aligned} L(\mathcal{X}|\boldsymbol{\theta}) &= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(\mathbf{X}_i - \boldsymbol{\mu}_i)^2}{2\sigma^2}) \\ &= (2\pi\sigma^2)^{-N/2} \exp(-\frac{\sum_{i=1}^{N}(\mathbf{X}_i - \boldsymbol{\mu}_i)^2}{2\sigma^2}). \end{aligned} \tag{2.12}$$

Thus the maximizer of the model likelihood $L(\boldsymbol{\theta})$ minimizes $\sum_{i=1}^{N}(\mathbf{X}_i - \boldsymbol{\mu}_i)^2$; hence the solution with the least squared error is the maximizer of the likelihood for independent and identically distributed measurement errors.

Maximum likelihood and maximum *a posteriori* estimators

The most likely estimate given experimental data is known as the **maximum likelihood estimator** (MLE). The MLE, denoted $\hat{\boldsymbol{\theta}}$ is typically computed via optimization of the likelihood function $L(\boldsymbol{\theta}) = P(D|\boldsymbol{\theta})$, i.e. $\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\text{argmax}}\ L(\boldsymbol{\theta})$, for data D.

The **sensitivities** of the likelihood function are the gradients of the likelihood function with respect to the model parameters, i.e. $\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_1}, \ldots \frac{\partial L(\boldsymbol{\theta})}{\partial \theta_N}$. The gradients are useful for improving the performance of numerical optimization algorithms for obtaining the MLE. If the gradients cannot be obtained analytically, they are typically estimated numerically instead.

An alternative to the MLE is the **maximum a posteriori** estimator (MAP), which is the maximizer of the posterior probability given by (2.2), i.e. $\text{MAP} = \underset{\boldsymbol{\theta}}{\text{argmax}}\ \frac{P(D|\boldsymbol{\theta})P(\boldsymbol{\theta})}{P(D)}$. In the event of an "uninformative" prior, that is, $P(\boldsymbol{\theta}) = C$ (for some constant C), the MAP reproduces the MLE.

Confidence intervals

Both the MLE and the MAP represent *point estimates* of the parameter set most in agreement with the observed data, i.e., they represent single best (possibly multidimensional) point in the possible parameter space. However, one is usually also interested in the **confidence interval** (CI) of the parameters. That is, one would like to know the region of parameter space, for which there is a high probability that the parameters are contained within this region. The **significance level**, α is the probability that the (unknown) true value of parameter *does not* lie within this region. For example, one commonly looks at the 95% CI (significance $\alpha = 0.05$): the probability of the model parameters not being contained within this interval is estimated to be 5%.

In some cases, it is possible to derive the **asymptotic distribution** of the sample estimator, i.e. the distribution of the sample estimator converges to this distribution as the number of samples increases indefinitely. In other cases, the distribution of the estimator is known exactly regardless of the sample size. If the distribution of the estimator of

the parameter is known, then the CI can be straightforwardly computed. For a univariate parameter, the CI is given by the region $\theta_\ell \leq \theta \leq \theta_u$ such that $\Phi(\theta_\ell) = \frac{\alpha}{2}$ and $\Phi(\theta_u) = 1-\frac{\alpha}{2}$, where $\Phi(\theta)$ is the CDF of θ, given by (2.4). If the distribution is not known, then one can approximate the CI based on the empirical distribution, e.g. using a bootstrap estimator, see e.g. [160].

In higher dimensions, instead of a confidence interval one speaks of a multidimensional confidence region.

Profile likelihoods

If a parametric function for the likelihood is not known, it is possible to instead use **profile likelihoods** to estimate the region most likely to contain the correct parameters for a given model. For parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_d)$, the profile likelihood of parameter θ_k is computed by fixing the value of θ_k to a value c, while simultaneously maximizing the likelihood using the remaining parameters, $\boldsymbol{\theta} \setminus \theta_k$:

$$PL(\theta_k; c) = \max_{\boldsymbol{\theta}\setminus\theta_k} \ell(\theta_k = c; \boldsymbol{\theta} \setminus \theta_k) \tag{2.13}$$

Intuitively, the profile likelihood is an indication of the identifiability of a single model parameter θ_k: if the profile likelihood decreases in both directions away from the maximum likelihood estimator, it indicates that the likelihood cannot be increased by some perturbation to the other model parameters. In such a case θ_k is said to be identifiable. Conversely, if the profile likelihood is flat in either direction it indicates a lack of identifiability, since perturbing θ_k away from the MLE does not cause the likelihood to decrease. In other words, there are other parameter combinations which agree equally well with the data.

Bayesian parameter inference

Although profile likelihoods give an estimate of the region containing individual model parameters, they do not show how parameters estimates relate to each other. For example, model parameters which are not identifiable may exhibit strong correlations, which can give rise to manifolds in parameter space for which the likelihood does not change appreciably.

If the evidence integral $P(\mathbf{D}) = \int (\mathbf{D}|\boldsymbol{\theta})P(\boldsymbol{\theta})d\boldsymbol{\theta}$ is tractable, then it is possible to directly compute the posterior probability $P(\boldsymbol{\theta}|\mathbf{D})$, see (2.2). However, it is generally not possible to compute the normalizing factor $P(\mathbf{D})$ in closed form.

Markov chain Monte Carlo sampling

If the evidence integral cannot be computed directly, one may instead approximate the posterior distribution $P(\boldsymbol{\theta}|\mathbf{D}) = P(\mathbf{D}|\boldsymbol{\theta})P(\boldsymbol{\theta})/P(\mathbf{D})$ using Monte Carlo (MC) sampling. If one can generate a series of samples $\boldsymbol{\theta}^{(k)}, k = 1, \ldots, N$ from the posterior, then it can be approximated by a mixture of Dirac delta functions, centered on the N points:

$$P(\boldsymbol{\theta}|\mathbf{D}) \approx \sum_{k=1}^{N} \delta(\boldsymbol{\theta} - \boldsymbol{\theta}^{(k)}). \tag{2.14}$$

Thus with Monte Carlo sampling it is not necessary to compute the model evidence. The PDF of an arbitrary point not in the support of the Delta mixture can then be obtained by applying kernel density estimation to the sampled points.

To guarantee that the points $\boldsymbol{\theta}^{(k)}$ are sampled from the target density, they must be chosen in a particular way. Many techniques exist for drawing MC samples from a distribution, for example slice sampling, importance sampling, rejection sampling, etc., see e.g. [161]. However, if the normalizing factor $P(\mathbf{D})$ is not known, one may instead use **Markov chain Monte Carlo** (MCMC). MCMC works by constructing a "chain", i.e. a series of samples, which satisfy the Markov property, i.e. the distribution of the next sample depends only on the current sample, see Section 2.3.2. The Markov chain constitutes a random sequence in the parameter space. If the chain is carefully constructed, the distribution of samples in the chain will converge eventually to the target distribution, $P(\boldsymbol{\theta}|\mathbf{D})$.

The Markov chain is initialized to some value $\boldsymbol{\theta}_0$, and at each iteration i of the MCMC algorithm, a new sample $\boldsymbol{\theta}_{i+1}$ is generated from conditional distribution given the current sample, $P(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D})$. The goal is to construct a chain such that the distribution of a set of samples drawn from chain (after the chain has converged to its stationary distribution), is equal to the target density, i.e. the posterior density.

The MCMC algorithm consists of two stages:

1. proposal of a new parameter set $\boldsymbol{\theta}_{i+1}$ with probability density $P_{\text{prop}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i)$

2. acceptance or rejection of the new proposal according to an acceptance probability $P_{\text{accept}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D})$

Hence the probability of generating a new sample $\boldsymbol{\theta}_{i+1}$, conditional on the current state $\boldsymbol{\theta}_i$ is given by the product of these two probabilities: $P(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D}) = P_{\text{prop}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i) \times P_{\text{accept}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D})$. Thus, the marginal probability of $\boldsymbol{\theta}_{i+1}|\mathbf{D}$ is given by:

$$\begin{aligned} P(\boldsymbol{\theta}_{i+1}|\mathbf{D}) &= \int P(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D}) P(\boldsymbol{\theta}_i|\mathbf{D}) d\boldsymbol{\theta}_i \\ &= \int P_{\text{prop}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i) P_{\text{accept}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D}) P(\boldsymbol{\theta}_i|\mathbf{D}) d\boldsymbol{\theta}_i \end{aligned} \tag{2.15}$$

Assuming that the current state $\boldsymbol{\theta}_i$ is a sample from the target distribution, i.e. $P(\boldsymbol{\theta}_i|\mathbf{D}) = \frac{P(\mathbf{D}|\boldsymbol{\theta}_i)P(\boldsymbol{\theta}_i)}{P(\mathbf{D})}$, (2.15) becomes

$$P(\boldsymbol{\theta}_{i+1}|\mathbf{D}) = \int P_{\text{prop}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i) P_{\text{accept}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D}) \frac{P(\mathbf{D}|\boldsymbol{\theta}_i)P(\boldsymbol{\theta}_i)}{P(\mathbf{D})} d\boldsymbol{\theta}_i \tag{2.16}$$

If we propose new parameters $\boldsymbol{\theta}_{i+1}$ according to some arbitrary proposal function q such that $P_{\text{prop}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i) = q(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i)$, and accept with probability

$$P_{\text{accept}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D}) = \frac{P(\mathbf{D}|\boldsymbol{\theta}_{i+1})P(\boldsymbol{\theta}_{i+1})}{P(\mathbf{D}|\boldsymbol{\theta}_i)P(\boldsymbol{\theta}_i)} \frac{q(\boldsymbol{\theta}_i|\boldsymbol{\theta}_{i+1})}{q(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i)} \tag{2.17}$$

then the probability of the sample $\boldsymbol{\theta}_{i+1}$ given by (2.16) reduces to

$$\begin{aligned}
&\int P_{\text{prop}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i) P_{\text{accept}}(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i, \mathbf{D}) P(\boldsymbol{\theta}_i|\mathbf{D}) d\boldsymbol{\theta}_i \\
&\qquad = \int q(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i) \left[\frac{P(\mathbf{D}|\boldsymbol{\theta}_{i+1}) P(\boldsymbol{\theta}_{i+1})}{P(\mathbf{D}|\boldsymbol{\theta}_i) P(\boldsymbol{\theta}_i)} \frac{q(\boldsymbol{\theta}_i|\boldsymbol{\theta}_{i+1})}{q(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i)} \right] \frac{P(\mathbf{D}|\boldsymbol{\theta}_i) P(\boldsymbol{\theta}_i)}{P(\mathbf{D})} d\boldsymbol{\theta}_i \\
&\qquad = \int \frac{P(\mathbf{D}|\boldsymbol{\theta}_{i+1}) P(\boldsymbol{\theta}_{i+1})}{P(\mathbf{D})} q(\boldsymbol{\theta}_i|\boldsymbol{\theta}_{i+1}) d\boldsymbol{\theta}_i \\
&\qquad = \frac{P(\mathbf{D}|\boldsymbol{\theta}_{i+1}) P(\boldsymbol{\theta}_{i+1})}{P(\mathbf{D})} \int q(\boldsymbol{\theta}_i|\boldsymbol{\theta}_{i+1}) d\boldsymbol{\theta}_i \\
&\qquad = \frac{P(\mathbf{D}|\boldsymbol{\theta}_{i+1}) P(\boldsymbol{\theta}_{i+1})}{P(\mathbf{D})} = P(\boldsymbol{\theta}_{i+1}|\mathbf{D})
\end{aligned} \tag{2.18}$$

Hence, $\boldsymbol{\theta}_{i+1}$ is a sample from the target, posterior density.

Whenever the acceptance probability (2.17) is larger than 1, the new sample $\boldsymbol{\theta}_{i+1}|\mathbf{D}$ should be accepted with probability 1. Thus, by proposing according to $q(\boldsymbol{\theta}_{i+1}|\boldsymbol{\theta}_i)$, and accepting according to (2.17), the probability density of the samples generated is exactly that of the target density. The MCMC algorithm with the proposal and acceptance probabilities presented in this section is known as the **Metropolis-Hastings (MH) algorithm** [162].

The Markov chain is initialized from an arbitrary point $\boldsymbol{\theta}_0$, and thus does not represent a sample from the target density. Nonetheless, it is possible to show that the target distribution is an invariant distribution of the process defined by the MH algorithm. To see this, let $P^*(\boldsymbol{\theta})$ denote the invariant or steady-state distribution of $\boldsymbol{\theta}$, i.e. the distribution $P^*(\boldsymbol{\theta})$ is unchanged when performing the proposal and acceptance steps of the MH algorithm. For each step of the MH algorithm, the expected change in the density for a state $\boldsymbol{\theta}$ is given by the difference in flux into and out of that state from all other states $\boldsymbol{\theta}'$:

$$\mathbb{E}[\Delta P^*(\boldsymbol{\theta})] = \int P^*(\boldsymbol{\theta\prime}) P(\boldsymbol{\theta}|\boldsymbol{\theta\prime}) - P^*(\boldsymbol{\theta}) P(\boldsymbol{\theta\prime}|\boldsymbol{\theta}) d\boldsymbol{\theta}' \tag{2.19}$$

where $P(\boldsymbol{\theta}'|\boldsymbol{\theta}) = P_{\text{prop}}(\boldsymbol{\theta}'|\boldsymbol{\theta}) \times P_{\text{accept}}(\boldsymbol{\theta}'|\boldsymbol{\theta})$.

Thus, for the distribution to be unaffected by the MH sampling procedure, the expected change must be zero for each state, giving the so-called **detailed balance** equation:

$$P^*(\boldsymbol{\theta}') P(\boldsymbol{\theta}|\boldsymbol{\theta}') = P^*(\boldsymbol{\theta}) P(\boldsymbol{\theta}'|\boldsymbol{\theta}) \tag{2.20}$$

We can easily verify that (2.20) is satisfied by the proposal density $P_{\text{prop}}(\boldsymbol{\theta}'|\boldsymbol{\theta}) = q(\boldsymbol{\theta}'|\boldsymbol{\theta})$, and acceptance probability defined by (2.17), by substitution:

$$\begin{aligned}
P^*(\boldsymbol{\theta}') P(\boldsymbol{\theta}|\boldsymbol{\theta}') &= P^*(\boldsymbol{\theta}') q(\boldsymbol{\theta}|\boldsymbol{\theta}') P_{\text{accept}}(\boldsymbol{\theta}|\boldsymbol{\theta}') \\
&= P^*(\boldsymbol{\theta}') q(\boldsymbol{\theta}|\boldsymbol{\theta}') \frac{P(\mathbf{D}|\boldsymbol{\theta}) P(\boldsymbol{\theta}) q(\boldsymbol{\theta}'|\boldsymbol{\theta})}{P(\mathbf{D}|\boldsymbol{\theta}') P(\boldsymbol{\theta}') q(\boldsymbol{\theta}|\boldsymbol{\theta}')}
\end{aligned} \tag{2.21}$$

where we have assumed that the $P_{\text{accept}}(\boldsymbol{\theta}|\boldsymbol{\theta}') \leq 1$, hence $P_{\text{accept}}(\boldsymbol{\theta}'|\boldsymbol{\theta}) = 1$.

By setting $P^*(\boldsymbol{\theta}') = \frac{P(\mathbf{D}|\boldsymbol{\theta}')P(\boldsymbol{\theta}')}{P(\mathbf{D})}$ (2.21) reduces to

$$\begin{aligned} P^*(\boldsymbol{\theta}')P(\boldsymbol{\theta}|\boldsymbol{\theta}') &= \frac{P(\mathbf{D}|\boldsymbol{\theta}')P(\boldsymbol{\theta}')}{P(\mathbf{D})} q(\boldsymbol{\theta}|\boldsymbol{\theta}') \frac{P(\mathbf{D}|\boldsymbol{\theta})P(\boldsymbol{\theta})q(\boldsymbol{\theta}'|\boldsymbol{\theta})}{P(\mathbf{D}|\boldsymbol{\theta}')P(\boldsymbol{\theta}')q(\boldsymbol{\theta}|\boldsymbol{\theta}')} \\ &= \frac{P(\mathbf{D}|\boldsymbol{\theta})P(\boldsymbol{\theta})}{P(\mathbf{D})} q(\boldsymbol{\theta}'|\boldsymbol{\theta}) \\ &= P^*(\boldsymbol{\theta})q(\boldsymbol{\theta}'|\boldsymbol{\theta}) \\ &= P^*(\boldsymbol{\theta})P(\boldsymbol{\theta}'|\boldsymbol{\theta}) \end{aligned} \tag{2.22}$$

where the last line follows from the assumption (2.17). Thus, $P^*(\boldsymbol{\theta}) = \frac{P(\mathbf{D}|\boldsymbol{\theta})P(\boldsymbol{\theta})}{P(\mathbf{D})} = P(\boldsymbol{\theta}|\mathbf{D})$, and the invariant distribution of the MH algorithm is the posterior probability density of $\boldsymbol{\theta}$.

Convergence can be checked via a number of criteria, e.g. by checking for stationarity of the distribution of the chain over a sufficiently large interval using e.g. a t-test, or the Geweke spectral density diagnostic for convergence of a Markov chain [163]. Typically one discards the initial portion of the Markov chain since during this "burn-in" phase, the samples are not being drawn from the target distribution. Deciding which fraction of the samples to discard, however, is generally a matter of preference.

One final important consideration, is that due to the Markov nature of the sample generation, a strong degree of autocorrelation can be induced amongst generated samples. Thus, the chain is not guaranteed to produce samples that are statistically independent from one another and can lead to biased results for any function estimated from the Markov samples, as compared to the true target density. To counteract the high autocorrelation of the Markov chain, *thinning* is sometimes advocated, wherein only a fraction of the samples are retained such that the autocorrelation of the thinned samples is minimal. However, the duration of the burn-in phase, the width of the proposal function $q(\boldsymbol{\theta}'|\boldsymbol{\theta})$ and the degree of thinning generally must be carefully hand-tuned to get enough samples, limit the degree of autocorrelation, and ensure convergence to the steady state of the Markov chain.

2.3 Dynamical systems

It is quite obvious that more can be learned about a system from studying its time-dependent behavior than by studying information gleaned only at fixed time-points. We refer to a system that evolves in time as a **dynamical system**, or **process**. We further differentiate between deterministic and stochastic processes where the former is entirely predictable given initial conditions and parameters, and the latter must be described instead in terms of probability distributions.

2.3.1 Deterministic processes

Deterministic processes provide a framework for analyzing the time-dependent behavior of any system for which the "equations of motion" for arbitrary time-dependent quantities of the system are known.

In general, one may consider the instantaneous configuration of a system to constitute a "coordinate" in the state space of all possible configurations of the system, i.e. the so-called **state space**, and the evolution of the system constitutes a trajectory through this state space. In some cases it may be possible to write down an equation capturing the configuration of the system, x, at any time t. However, for many systems a closed-form solution for the configuration of the system is not possible.

Ordinary differential equations

While it may not be possible to derive the closed-form solution of the exact configuration of the system at all points in time, it is often possible to describe the *instantaneous change* in the system's configuration. Systems for which the rate of change depends on only the current state of the system obey **ordinary differential equations** (ODEs), written as:

$$\frac{dx}{dt} = f(x, t, \boldsymbol{\theta}) \tag{2.23}$$

where $f(x, t, \boldsymbol{\theta})$ is the instantaneous change of x, or **derivative** of x, and all relevant model parameters are contained in the vector $\boldsymbol{\theta}$. The configuration of the system at an arbitrary time can be computed via the integral $\int_0^t f(x, s, \boldsymbol{\theta})ds$ either analytically if possible, or numerically otherwise.

Importantly, the evolution of a system defined by an ODE depends only on the instantaneous configuration of the system, thus it is Markovian. If the system depends on its past configuration, its evolution must either be modeled using a different mathematical framework, such as *delayed differential equations*, or the state space must be expanded to incorporate past configurations of the system as additional dimensions, upon which the Markov property is restored.

Sensitivity analysis

For the purposes of model fitting and parameter inference, one is often interested in knowing how the solution of the model depends on the model parameters. This dependence is captured by the rate of change of the state per change in the model parameters, and is known as the **sensitivity**. If $x(t)$ is a (univariate) deterministic process, given by the ODE $\frac{dx}{dt} = f(x, t, \boldsymbol{\theta})$, where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)^T$ is a vector of model parameters influencing the equations of motion, then the sensitivity of $x(t)$ with respect to the k^{th} model parameter is given by:

$$S_k = \frac{\partial x}{\partial \theta_k}. \tag{2.24}$$

Eq. (2.24) captures the variation in $x(t)$ in response to perturbation to parameters $\boldsymbol{\theta}$ about a point of interest; thus it represents a "local" sensitivity analysis. Alternatively, the variation in $x(t)$ can be evaluated over a range of allowable parameter values of $\boldsymbol{\theta}$ (e.g. using Monte Carlo sampling), a strategy referred to as "global" sensitivity analysis [164].

Sensitivity equations If $x(t)$ can be computed analytically, i.e. if the integral $\int_0^t f(x, s, \boldsymbol{\theta})ds$ is known, then the sensitivity (2.24) can be directly computed by partial differentiation.

However, since the integral is often unknown, one typically computes the sensitivities numerically instead. It is straightforward to see that by differentiating the ODE with respect to the model parameters, one arrives at a new set of ODEs for each of the sensitivities:

$$\begin{aligned} \frac{\partial}{\partial\theta_k}\frac{dx}{dt} &= \frac{d}{dt}\frac{\partial x}{\partial\theta_k} \\ &= \frac{d}{dt}S_k \\ &= \frac{\partial}{\partial\theta_k}f(x(t),t,\boldsymbol{\theta}) \\ \frac{d}{dt}S_k &= \frac{\partial f}{\partial x}\underbrace{\frac{\partial x}{\partial\theta_k}}_{S_k}+\frac{\partial f}{\partial\theta_k} \end{aligned} \tag{2.25}$$

where the last line follows from the chain rule of differentiation. Thus, in addition to integrating the equations of motion (2.23) for the state of the system, one integrates the system of ODEs (2.25) as well in order to obtain the sensitivities with respect to model parameters. Higher order sensitivities, e.g. $\frac{\partial^2}{\partial\theta_k\partial\theta_l}\frac{dx}{dt}$, are defined analogously and obtained by further partial differentiation. Similarly, sensitivities for a multivariate variable $\mathbf{x} = (x^1, x^2, \ldots, x^d)^T$ with derivative $\dot{\mathbf{x}} = f(\mathbf{x}, t, \boldsymbol{\theta}) = (f^1, f^2, \ldots, f^d)^T$ are defined componentwise as:

$$\frac{d}{dt}S_k^i = \sum_{j=1}^{d}\frac{\partial f^i}{\partial x^j}\frac{\partial x^j}{\partial\theta_k}+\frac{\partial f^i}{\partial\theta_k} \tag{2.26}$$

with $S_k^i = \frac{\partial x^i}{\partial\theta_k}$.

Finite difference approximations If the sensitivity equations cannot be computed, or if the integration of the sensitivity equations is prohibitively costly, one may instead utilize the **finite difference approximation** in order to estimate the sensitivities (2.24).

The *forward finite difference* of a function $f(x)$ is computed via the Taylor expansion about the point x:

$$f(x+h) = f(x) + hf'(x) + \frac{1}{2}h^2f''(x) + \ldots \tag{2.27}$$

where $f'(x) = \frac{d}{dx}f(x)$.

Rearranging (2.27), we see

$$f'(x) = \frac{f(x+h)-f(x)}{h} + \mathcal{O}(h). \tag{2.28}$$

Thus $f'(x)$ can be approximated by the difference (2.28), if the perturbation h is sufficiently small, for which higher order terms are negligible.

A more accurate approximation of given by the *central finite difference approximation* as follows:

$$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{1}{2}h^2 f''(x) + \dots \\ f(x-h) &= f(x) - hf'(x) + \frac{1}{2}h^2 f''(x) + \dots \\ f(x+h) - f(x-h) &= 2hf'(x) + \dots \\ f'(x) &= \frac{f(x+h) - f(x-h)}{2h} + \mathcal{O}(h^2) \end{aligned}$$

The finite difference approximation to higher order derivatives is computed analogously. The central difference approximation to the second derivative of a function $f(x, y)$ is computed as:

$$\frac{\partial^2 f}{\partial x \partial y} \approx \frac{f(x+h, y+k) - f(x+h, y-k) - f(x-h, y+k) - f(x-h, y-k)}{4hk} \tag{2.29}$$

for some small perturbations h and k. We note that the finite difference approximation is generally inferior to the sensitivities obtained via integration of the sensitivity equations (2.25), see e.g. [165] for discussion.

2.3.2 Stochastic processes

In many systems, it is not possible to predict the time-dependent behavior exactly, due to either lack of sufficient knowledge of the system, or due to the inherent randomness, e.g. arising from quantum mechanical effects. For such systems, it is necessary instead to model the behavior probabilistically using knowledge of the system's statistical features. Systems for which the configuration evolves according to probabilistic rules are known as **stochastic processes**.

Markov processes

The simplest example of a stochastic process is the so-called **Markov process.** Markov processes obey the *Markov property,* namely, that the future state of the system depends only on the current state of the system. For this reason, Markov processes are said to be *memoryless.* Consider a stochastic process whose state at time t is denoted by X_t. The configuration of the system at a series of timepoints $t_1, t_2, \dots, t_N$ is denoted by X_{t_1}, X_{t_2}, etc. The Markov property is thus equivalent to the statement

$$P(X_{t_i} | X_{t_1}, X_{t_2}, \dots, X_{t_{i-1}}) = P(X_{t_i} | X_{t_{i-1}}). \tag{2.30}$$

That is, the probability density of the system at the next timepoint t_{i+1} is conditionally independent of the configuration of the system at all timepoints prior to t_i.

Due to the Markov property (2.30), Markov processes have the additional convenient feature that the joint probability density of the series $X_{t_1}, X_{t_2}, \dots$ can be factorized as

$$\begin{aligned} P(X_{t_1}, X_{t_2}, \dots, X_{t_N}) &= P(X_{t_1}) P(X_{t_2} | X_{t_1}) P(X_{t_3} | X_{t_2}) \dots P(X_{t_N} | X_{t_{N-1}}) \\ &= P(X_{t_1}) \prod_{i=2}^{N} P(X_{t_i} | X_{t_{i-1}}) \end{aligned} \tag{2.31}$$

Higher order Markov processes The Markov property (2.30) is in general an abstraction of a physical process. In reality, almost all systems possess some "memory", and thus a Markov process is generally only approximately valid. However, in many cases it may be possible to restore the Markov property by expanding the configuration of the system to include the past states of the system as additional dimensions in the state space. Similarly, one may define a *higher-order Markov process* to be a Markov process that depends not just on the current configuration of the system, but also on some number N of the previous states of the system. For example a second-order Markov process has a probability density

$$P(X_{t_i}|X_{t_0},\ldots,X_{t_{i-1}}) = P(X_{t_i}|X_{t_{i-1}},X_{t_{i-2}}) \tag{2.32}$$

Markov jump processes So far the configuration of the system X was not rigorously defined. If the state space is a subset of the d-dimensional integer lattice, $\Omega \subseteq \mathbb{Z}^d$, then the system may only change by discrete quantities, or *jumps*, within the state space. Hence, the Markov process is said to be a *Markov jump process* (MJP). One common application of MJPs is in the context of chemical physics, where the number of molecules of each chemical reactant is of course non-negative integer-valued, and the probabilistic evolution of the system depends only on the current configuration, thus obeying the Markov property.

For MJPs, one may define a vector of probabilities, $\mathbf{P}(t_i)$, comprised of the probabilities for each point in the integer lattice corresponding to the state space of the system at time t_i. For example, for a one-dimensional state space with minimal value 0 and maximal value N, $\mathbf{P}(t_i) = (P(X_{t_i}=0), P(X_{t_i}=1),\ldots,P(X_{t_i}=N))^T$.

For such a finite-dimensional MJP, the evolution of the system can be described by the matrix-vector equation:

$$\mathbf{P}(t_{i+1}) = \mathbf{W}(t)\cdot\mathbf{P}(t_i) \tag{2.33}$$

where $\mathbf{W}(t)$ is the *stochastic matrix* of the process, and is defined as

$$[\mathbf{W}(t)]_{kl} = P(X_{t_{i+1}}=k|X_{t_i}=l) \tag{2.34}$$

that is, the $(k,l)^{th}$ entry of $\mathbf{W}$ corresponds to the probability of transitioning from the state l at time t_i to the state k, at time t_{i+1}. For this reason, $\mathbf{W}(t)$ is also known as the *transition matrix* of the process. If $\mathbf{W}(t) = \mathbf{W}$ for all times, then the process is said to be **homogeneous**.

Continuous time Markov chains In Section 2.3.2, we defined a series of times $t_i, i = 1,\ldots,N$ for which the configuration of the system was evaluated. Such a discrete time setting is natural for some systems. For example the configuration of a turn-based game like chess might be modeled using a discrete time Markov process, where the "times" correspond to the turn number. However, many systems do not possess a natural, discrete time scale but rather evolve continuously through time. A Markov process defined over a continuous time variable t is said to be a continuous time Markov process, or **continuous time Markov chain** (CTMC). The CTMC evolves probabilistically, and the probability of the system realizing configuration x at time t is denoted $P(X_t = x)$. Its evolution is given by the **transition rate matrix Q**: $\frac{d}{dt}\mathbf{P}(t) = \mathbf{Q}(t)\cdot\mathbf{P}(t)$.

Properties of stochastic processes

Steady state A stochastic process is said to be in **steady state** if the probability density describing the system is time-invariant:

$$P(x_t) = P(x_{t+\tau}), \forall \tau \Leftrightarrow \frac{\partial P(x_s)}{\partial s} = 0 \tag{2.35}$$

For a homogeneous MJP, this implies $\mathbf{Q} \cdot \mathbf{P}_{SS}(t) = 0$. A stochastic process is **stationary** if the distribution of a sequence of samples from the process is invariant for arbitrary time-shifts.

Moments The value of the random variable X_t at time t is only known probabilistically. Its moments are defined in analogously to the case of a simple experiment, see Section 2.2.4. Briefly, the expectation of a univariate X_t following a stochastic process is defined as:

$$\mathbb{E}[X_t] = \int x \; P(X_t = x) dx. \tag{2.36}$$

and the N^{th}-order moment by

$$M_N = \mathbb{E}[(X_t)^N] = \int x^N \; P(X_t = x) dx. \tag{2.37}$$

This is generalized to the N^{th}-order **cross-moment** by

$$\mathbb{E}[X_{t_1} X_{t_2} \dots X_{t_N}] = \int x_1 x_2 \dots x_N P(X_{t_1} = x_1, X_{t_2} = x_2, \dots, X_{t_N} = x_N) \, dx_1 \dots dx_N \tag{2.38}$$

The moments of discrete valued X are given analogously, using summation over the state space of X instead of integration.

Moment generating function The **moment generating function** (MGF) of a discrete valued stochastic process X_t is defined as:

$$G(Z,t) = \mathbb{E}[Z^{X_t}] = \sum_{x=0}^{\infty} P(X_t = x) Z^x \tag{2.39}$$

using an auxillary variable Z. The probability of a particular state $P(X_t = n)$ can be recovered by differentiating (2.39) n times with respect to Z, and evaluating the resulting derivative for $Z = 0$:

$$\begin{aligned} \frac{\partial^n}{\partial Z^n} G(Z,t) &= \sum_{x=0}^{\infty} P(X_t = x) x(x-1) \dots (x-n+1) Z^{x-n} \\ &= \sum_{x=0}^{\infty} P(X_t) \frac{x!}{n!} Z^{x-n} \end{aligned} \tag{2.40}$$

$$\begin{aligned} \frac{\partial^n}{\partial Z^n} G(0,t) &= \sum_{x=0}^{\infty} P(X_t = x) \frac{x!}{n!} 0^{x-n} \\ &= P(X_t = n) \frac{n!}{n!} = P(X_t = n) \end{aligned} \tag{2.41}$$

Here only the $x = n$ term of the sum remains after setting $Z = 0$.

To compute the N^{th} central moment of the process, (2.37), first change the auxiliary variable Z such that $Z = e^k$, for which the MGF becomes

$$\begin{aligned} G(k,t) &= \mathbb{E}[e^{kX}] \\ &= \sum_{x=0}^{\infty} P(X_t = x) e^{kx}. \end{aligned} \tag{2.42}$$

Differentiating $G(k,t)$ N times with respect to k, and evaluating at $k = 0$ yields:

$$\begin{aligned} \frac{\partial^N}{\partial k^N} G(k,t) &= \frac{\partial^N}{\partial k^N} \sum_{x=0}^{\infty} P(X_t = x) e^{kx} \\ &= \sum_{x=0}^{\infty} P(X_t = x) x^N e^{kx} \\ \frac{\partial^N}{\partial k^N} G(0,t) &= \sum_{x=0}^{\infty} P(X_t = x) x^N \\ &= M_N \end{aligned} \tag{2.43}$$

Autocorrelation and cross-correlation The **autocorrelation** of a univariate stochastic process X_t at times s and $s + \tau$ is defined as the function

$$R(s, s+\tau) = \frac{\mathbb{E}[(X_s - \mathbb{E}[X_s])(X_{s+\tau} - \mathbb{E}[X_{s+\tau}])]}{\sqrt{\text{var}[X_s]\,\text{var}[X_{s+\tau}]}}. \tag{2.44}$$

In the literature, the term "autocorrelation" is sometimes used to denote the quantity in (2.44) without the normalizing factors in the denominator.

The function $R(s, s+\tau)$ measures the degree to which fluctuations in the stochastic process at time s are dependent on fluctuations at a time lag of τ. If the process is stationary such that the mean μ and variance $\text{var}[X_s] = \sigma^2$ are time-invariant, then (2.44) depends only on the time shift τ

$$R(\tau) = \frac{\mathbb{E}[(X_s - \mu)(X_{s+\tau} - \mu)]}{\sigma^2}. \tag{2.45}$$

For multivariate systems, the **cross-correlation** between two random variables X and Y, observed at times s and $s + \tau$ respectively is define analogously to (2.44) as:

$$R_{XY}(s, s+\tau) = \frac{\mathbb{E}[(X_s - \mu_X(s))(Y_{s+\tau} - \mu_Y(s+\tau))]}{\sqrt{\text{var}[X_s]\,\text{var}[Y_{s+\tau}]}}. \tag{2.46}$$

Ergodicity A stationary stochastic process is said to be **ergodic** if its time average is identical to its expectation:

$$\begin{aligned} \lim_{\Delta t \to \infty} \frac{1}{\Delta t} \int_t^{t+\Delta t} X_{t'}\, dt' &\overset{!}{=} \mathbb{E}[X] \\ &= \int x P(X_t = x) dx \end{aligned} \tag{2.47}$$

requiring that the expectation $\mathbb{E}[X]$ be time-invariant, i.e. the process is stationary. Intuitively, if the process is ergodic, it implies that the statistical features (moments, etc.) of the process may be estimated by collecting sufficiently many samples from a single realization of the process instead of through an ensemble of replicates (i.e. repeated measurements under identical conditions).

2.4 Chemical physics

2.4.1 Chemical reaction networks

Cells must continually perform chemical reactions in order to meet their metabolic needs and process signals from the extracellular environment. Thus, the cell can be described in terms of its time-varying chemical composition which may evolve according to physicochemical interactions subsuming all cellular activities such as enzymatic reactions for cellular metabolism, post-translational modification of proteins in signal transduction, binding of transcription machinery to DNA, etc. To model the possible interactions of a system of chemical species, we use **chemical reaction networks**, which constitute a framework for concisely describing a set of possible reactions between educts (consumed during a reaction) and products (produced), for each reaction.

Definition

A specific state of the intracellular chemical system is specified by the instantaneous number of molecules of each chemical species, e.g. transcripts of particular genes, proteins, transcription factors, or metabolites. The state of the system is described by a vector $\mathbf{X} = (x_1, x_2, \ldots, x_N)^T \in \mathbb{N}_0^{N_s}$ consisting of the copy numbers of all N_s relevant chemical species. Due to the discrete nature of molecules, the cellular state can only change by integer amounts, coinciding with the set of possible chemical reactions that may take place.

In particular, the j^{th} chemical reaction, denoted R_j, has associated to it a "net change vector", $\boldsymbol{\nu}_j \in \mathbb{Z}^{N_s}$, such that one firing of reaction j changes the system state by $\boldsymbol{\nu}_j$:

$$R_j : \mathbf{X} \to \mathbf{X} + \boldsymbol{\nu}_j \tag{2.48}$$

The **stoichiometric matrix**,

$$\mathbf{S} = \begin{bmatrix} \boldsymbol{\nu}_1 & \boldsymbol{\nu}_2 & \ldots & \boldsymbol{\nu}_R \end{bmatrix} \tag{2.49}$$

is the matrix composed from the change vectors of all reactions $j = 1 \ldots R$. The $(i,j)^{th}$ entry of the stoichiometric matrix gives the net change in species i for one firing of R_j. If we define a vector $\mathbf{n} = (n_1, \ldots, n_R)^T$ to be the number of times each reaction fires, then the product $\mathbf{S} \cdot \mathbf{n}$ is the total change in $\mathbf{X}$: $\mathbf{X} \to \mathbf{X} + \mathbf{S} \cdot \mathbf{n}$.

Reaction rates

Each reaction may take place at any point in time, providing that sufficiently many molecules of the educt are present for the reaction to take place. The probability for

each reaction to take place depends on the instantaneous quantity of the educts and on kinetic constants which are physical properties of the interacting molecules and the reaction environment, such as temperature.

Stochastic treatment Consider a reaction taking place between molecules of two hypothetical species S_1 and S_2 with abundances of x_1 and x_2 molecules, respectively. It can be derived from a microscopic physics argument, under the assumption of thermal equilibrium, that the probability of collision (and thus potentially of reaction) depends on the probability that S_2 lies within the *collision volume* of molecule S_1 [117]. The collision volume can be interpreted as the volume of the subspace in which the spatial positions of the two molecules overlap. The collision volume of S_1 depends on the sum of the two molecular radii $(r_1 + r_2)$, the instantaneous velocity v_{12} along the intermolecular axis $\vec{r}_{12}$, and the infinitesimal time window under consideration Δt. The probability of S_2 lying within this volume, assuming the reaction volume to be "well-mixed", is simply proportional to the fractional volume of the collision volume. Computing $\overline{v_{12}}$, the expectation over the velocity v_{12} at thermal equilibrium, and thus of a Maxwellian velocity distribution, leads to the following estimate of collision probability:

$$P_{coll}(\mathbf{X} = \mathbf{x}, \Delta t) = x_1 x_2 \Omega^{-1} \pi (r_1 + r_2)^2 \overline{v_{12}} \Delta t \tag{2.50}$$

where Ω is the reaction volume, and $\overline{v_{12}} = \left(\frac{8 k_B T}{\pi} \frac{m_1 + m_2}{m_1 * m_2}\right)^{1/2}$ the thermal average velocity, which depends on the molecular masses m_1 and m_2 of S_1 and S_2, respectively, the Boltzmann constant k_B, and temperature.

The total probability of a reaction occurring within a time interval of length Δt is thus the probability of a collision occurring within Δt times the probability of a reaction occurring following collision, which is given by a chemical kinetic constant specific to each reaction, i.e. $P_{react}(\mathbf{x}, \Delta t) = k \cdot P_{coll}(\mathbf{x}, \Delta t)$. The **propensity** of a reaction is the instantaneous probability of that reaction occurring, denoted $a(\mathbf{x}) = \lim_{\Delta t \to 0} P_{react}(\mathbf{x}, \Delta t)$. Note that the assumption of thermal equilibrium in Eq. (2.50) is not essential; for any non-equilibrium velocity distribution the appropriate mean relative velocity $\overline{v_{12}}$ can be computed and substituted. Equilibrium is assumed here for clarity of exposition.

Typically only three possible rate laws are considered, covering the cases of zeroth-order, unimolecular, and bimolecular reactions; reactions involving three or more species have negligibly small probability and are thus broken into a series of bimolecular reactions. The rate laws appropriate for each of the three cases are summarized in Table 2.1. In each case, the parameter k serves as the kinetic constant, which depends on the chemical nature of the reaction educts, as well as environmental properties such as pressure and temperature. The constant Ω describes the system reaction volume (e.g. the cellular volume, or nuclear volume); intuitively, second order reactions proceed at a rate inversely proportional to the system volume, as it becomes increasingly unlikely that the molecules will encounter one another by chance. The fourth case is that of a bimolecular reaction involving two molecules of the same species with copy number X, for which $\frac{1}{2}X(X-1)$ possible *distinct* reaction pairs exist:

The vector of molecular copy numbers at time t, $\mathbf{X}_t$, is a random variable described by a stochastic process, in particular by a continuous time Markov chain, see Section 2.3.2.

Table 2.1: Stochastic Reaction Rate Laws

Reaction Order	Example	Propensity Function (a)
Zeroth	$\emptyset \xrightarrow{k} S_1$	$k\,\Omega$
First	$S_1 \xrightarrow{k} S_2$	$k\,x_1$
Second (different species)	$S_1 + S_2 \xrightarrow{k} S_3$	$k\,x_1\,x_2\,\Omega^{-1}$
Second (same species)	$2S_1 \xrightarrow{k} S_2$	$\frac{k}{2}\,x_1\,(x_1-1)\Omega^{-1}$

Deterministic treatment Chemical systems consist of discrete numbers of molecules and are intrinsically noisy due to the underlying stochastic processes which dictate their probabilistic evolution. However, they may be approximated by deterministic processes under certain conditions. If the number of molecules is large, one may approximate the molecular copy number of each species by a molecular number density, or **concentration**, defined as $\boldsymbol{\phi} = \mathbf{X}/\Omega$, where Ω is the reaction volume.

When the number of molecules is sufficiently large, the propensity functions in Table 2.1 do not change substantially in a small time window Δt. Thus, the probability of each of the reactions occurring is approximately constant over the time interval Δt, and therefore the waiting times for each reaction firings are given by an exponential distribution (see Section A.3) with parameter $\lambda = a(\mathbf{X})\Delta t$, for a propensity function $a(\mathbf{X})$. The number of times R_j fires in this time interval, N_j, is therefore Poisson-distributed (see Section A.4):

$$\begin{aligned} N_j &\sim \text{Poiss}(a_j(\mathbf{X})\Delta t) \\ P(N_j = n_j|\mathbf{X}, \Delta t) &= \frac{(a_j(\mathbf{X})\Delta t)^{n_j} e^{-a_j(\mathbf{X})\Delta t}}{n_j!} \end{aligned} \tag{2.51}$$

where $a_j(\mathbf{X})$ is the reaction propensity for reaction j, as in Table 2.1. Moreover, the expectation of N_j is $\mathbb{E}[N_j] = a_j(\mathbf{X})\Delta t$. Each firing of reaction j changes the molecular copy number vector $\mathbf{X}$ by the change vector $\boldsymbol{\nu}_j$: $\Delta\mathbf{X} = \boldsymbol{\nu}_j N_j$. Thus the expected change in the molecular copy number vector in time Δt is given by:

$$\begin{aligned} \mathbb{E}[\Delta\mathbf{X}] &= \sum_j \boldsymbol{\nu}_j \mathbb{E}[N_j] \\ &= \mathbf{S}\cdot\mathbb{E}[\mathbf{N}] \\ &= \mathbf{S}\cdot\mathbf{a}(\mathbf{X})\Delta t \\ \frac{\mathbb{E}[\Delta\mathbf{X}]}{\Delta t} &= \mathbf{S}\cdot\mathbf{a}(\mathbf{X}) \end{aligned} \tag{2.52}$$

where $\mathbf{S}$ is the stoichiometric matrix (see Section 2.4.1), $\mathbf{N} = (N_1, N_2, \ldots, N_{N_s})^T$, and $\mathbf{a}(\mathbf{X}) = (a_1(\mathbf{X}), a_2(\mathbf{X}), \ldots, a_{N_s}(\mathbf{X}))^T$. Finally, we see that by dividing by the reaction volume Ω we obtain

$$\frac{\mathbb{E}[\Delta\mathbf{X}/\Omega]}{\Delta t} = \frac{\mathbb{E}[\Delta\boldsymbol{\phi}]}{\Delta t} = \mathbf{S}\cdot\frac{\mathbf{a}(\mathbf{X})}{\Omega} \tag{2.53}$$

In the limit of infinitesimal time,

$$\lim_{\Delta t\to 0}\frac{\mathbb{E}[\Delta\boldsymbol{\phi}]}{\Delta t} = \frac{d}{dt}\mathbb{E}[\boldsymbol{\phi}] = \mathbf{S}\cdot\frac{\mathbf{a}(\mathbf{X})}{\Omega} \tag{2.54}$$

Thus, the average concentrations evolve according to the ODE (2.54), with rates given by the volume-normalized reaction propensities, summarized for elementary reactions (i.e. at most bimolecular) in Table 2.2, using the convention $\phi_i = \frac{x_i}{\Omega}$, etc., for the concentration of species S_i. This ODE is known as the **reaction rate equation** (RRE) or **macroscopic rate equation** (MRE). Moreover, deterministic rate laws of the form in Table 2.2, i.e. the product of the concentration of each reaction educt exponentiated to its respective coefficient, are said to follow the **law of mass action.** The macroscopic rates correspond exactly to the renormalized stochastic reaction propensities in Table 2.1, except for the bimolecular reaction involving two molecules of the same species, for which the approximation $x_i(x_i - 1) \approx (x_i)^2$ is used.

The deterministic approximation is valid in the limit of large molecule numbers. Specifically, the number of firings for each reaction is Poisson-distributed (2.51), and thus the variance of the number of firings is equal to the mean, $\mathrm{var}(N_j|\mathbf{X}, \delta t) = a_j(\mathbf{X})\Delta t$. Hence, the "noise", defined as the ratio of the standard deviation to the mean number of reaction firings, $\eta_j = \frac{\sqrt{\mathrm{var}[N_j]}}{\mathbb{E}[N_j]} = \frac{1}{\sqrt{a_j(\mathbf{X})\Delta t}}$ tends to zero as the propensity $a_j(\mathbf{X})$ grows. Since the propensity for all first and second order reactions grows proportionally to the concentration (Table 2.2), the noise shrinks accordingly. For zeroth order reactions, where the propensity is independent of the number of molecules, the approximation is valid whenever the chemical kinetic constant k is sufficiently high, since $\eta \propto k^{-1/2}$.

Table 2.2: Deterministic Rate Laws for Elementary Reactions

Reaction Order	Example	Propensity Function (a)
Zeroth	$\emptyset \xrightarrow{k} S_1$	k
First	$S_1 \xrightarrow{k} S_2$	$k\,\phi_1$
Second (different species)	$S_1 + S_2 \xrightarrow{k} S_3$	$k\,\phi_1\phi_2$
Second (same species)	$S_1 + S_1 \xrightarrow{k} S_2$	$k\,\phi_1^2$

2.4.2 Chemical master equation

A system of interacting chemical species can take on a potentially infinite number of configurations, depending on constraints or conservation relations which might limit the possible state space. Ignoring the spatial arrangement of molecules within the system, the state of the system can be characterized by the molecular copy number vector, denoted $\mathbf{X}$, as above. The probability of the system occupying the state $\mathbf{x}$ at time t can be written as $P(\mathbf{X} = \mathbf{x}, t)$. The state of the system can only be changed via the "firing" of various reactions of the CRN, and hence $\mathbf{x}$ be reached from any feasible neighboring state $\mathbf{x} - \boldsymbol{\nu}_j$, by exactly one firing of reaction j with associated state change vector $\boldsymbol{\nu}_j$. Thus, the probability density of state $\mathbf{x}$, $P(\mathbf{x}, t)$, can change via flux in to and out of the state $\mathbf{x}$. The probability flux $J_{\mathbf{x},\mathbf{x}-\boldsymbol{\nu}_j}(t)$ from state $\mathbf{x} - \boldsymbol{\nu}_j$ to state $\mathbf{x}$, in an infinitesimal time interval Δt, is proportional to the probability of the system being in state $\mathbf{x} - \boldsymbol{\nu}_j$ and to the probability of the reaction R_j firing within time interval Δt given that the system is in that state: $J_{\mathbf{x},\mathbf{x}-\boldsymbol{\nu}_j}(t) = P(\mathbf{x}-\boldsymbol{\nu}_j, t)a_j(\mathbf{x}-\boldsymbol{\nu}_j)\Delta t$. Conversely, the firing of reaction j when the system is in state $\mathbf{x}$ results in flux to the state $\mathbf{x} + \boldsymbol{\nu}_j$: $J_{\mathbf{x}+\boldsymbol{\nu}_j,\mathbf{x}}(t) = P(\mathbf{x}, t)a_j(\mathbf{x})\Delta t$.

The total change within Δt to the probability density of state $\mathbf{x}$, at time t is given by:

$$\Delta P(\mathbf{x},t) = \sum_{j=1}^{R} \left[\underbrace{P(\mathbf{x}-\boldsymbol{\nu}_j,t)a_j(\mathbf{x}-\boldsymbol{\nu}_j)}_{J_{\mathbf{x},\mathbf{x}-\boldsymbol{\nu}_j}} - \underbrace{P(\mathbf{x},t)a_j(\mathbf{x})}_{J_{\mathbf{x}+\boldsymbol{\nu}_j,\mathbf{x}}} \right] \Delta t \tag{2.55}$$

for a CRN with R reactions in total. In the limit $\Delta t \to 0$, we obtain the so-called **chemical master equation** (CME):

$$\frac{\partial}{\partial t} P(\mathbf{x},t) = \sum_{j=1}^{R} P(\mathbf{x}-\boldsymbol{\nu}_j,t)a_j(\mathbf{x}-\boldsymbol{\nu}_j) - P(\mathbf{x},t)a_j(\mathbf{x}) \tag{2.56}$$

Equation (2.56) describes the probabilistic evolution of the random variable $\mathbf{X}$, and depends on the instantaneous configuration of the system and the reaction propensities $a_j(\mathbf{X}), j = 1,\ldots,R$.

If there is only one chemical species, then we can construct a vector

$$\mathbf{P}(t) = (P_0(t), P_1(t) \ldots, P_{N_{\max}}(t))^T$$

where $P_k(t) = P(\mathbf{X} = k, t)$, describing the probability of each possible molecular copy number at time t. The quantity $N_{\max}$ is potentially infinite depending on the constraints of the system, see Section 2.3.2. The rate of change of the probability density of the k^{th} state, is given by:

$$\frac{\partial}{\partial t} P_k(t) = \sum_{l=1}^{N_{\max}} \sum_{j=1}^{R} a_{jl}(t) P_l(t) \delta(k - l - \nu_j) - a_{jk}(t) P_k(t)$$

where $a_{jl}(t)$ is the reaction propensity of reaction j when the system is in state l. Thus, the rate of change of the entire vector $\mathbf{P}(t)$ is given by the simple linear equation:

$$\frac{\partial}{\partial t} \mathbf{P}(t) = \mathbf{A}(t) \cdot \mathbf{P}(t) \tag{2.57}$$

where the propensities matrix $\mathbf{A}$ is given by

$$[\mathbf{A}(t)]_{kl} = \begin{cases} \sum_{j=1}^{R} a_{jl}(t)\delta(k - \nu_j - l), & k \neq l \\ -\sum_{j=1}^{R} a_{jk}(t), & k = l \end{cases}.$$

Thus for a single species, the CME takes the form of a simple matrix-vector multiplication, and can be solved analytically if the dimensionality of the system is finite:

$$\mathbf{P}(t) = e^{\int A(t)dt} \mathbf{P}(0) \tag{2.58}$$

or $\mathbf{P}(t) = e^{\mathbf{A}t}\mathbf{P}(0)$, if $\mathbf{A}(t) = \mathbf{A}$.

For a system with $N_s > 1$ species, the probability density of the system is described by a tensor, rather than a vector. Similarly to (2.57), a linear ODE can be defined for the evolution of the probability density tensor over a subspace of the lattice of all possible copy numbers of each species. Hence, it becomes increasingly difficult to solve the CME directly via e.g. numerical integration as the dimension of the state space increases. In most cases, no analytical solution is known for the chemical master equation. Thus one either needs to numerically integrate (2.56), or use approximations to improve tractability.

Finite state projection A method known as the Finite State Projection (FSP) provides a good approximation to the solution of the CME, even when the number of molecules is potentially unbounded [166]. If the CRN is allowed to take on states $X \in \Omega$ (usually $\Omega \subseteq \mathbb{N}_0^d$ for a d-dimensional system), then the FSP defines a subspace $\widehat{\Omega} \subseteq \Omega$ which is presumed to contain the majority of the probability density of the system. With this subset, the submatrix $\widehat{\mathbf{A}}$ of the propensity matrix $\mathbf{A}$ corresponding to $\widehat{\Omega}$ is computed simply be retaining the rows and columns corresponding to the states in $\widehat{\Omega}$. Then, the (approximate) solution of the CME at a future timepoint is computed by computing the matrix exponential of $\widehat{\mathbf{A}}$, as in (2.58). The approximate probability of any state in $\widehat{\Omega}$ is obtained simply by solving the ODE using the truncated transition matrix $\widehat{\mathbf{A}}$. The error of the FPS can be ascertained by summing the probability density of the approximate solution to the CME, and taking the difference from 1 (the total density should of course be 1). If the error is too great, the approximate state space $\widehat{\Omega}$ should be expanded. The FSP has the advantage of being simple to compute (as long as $\widehat{\Omega}$ is reasonably small), and the error can easily be computed. However, it may become difficult to predict which $\widehat{\Omega}$ provides an adequate approximation *a priori*, which might lead to additional computational overhead as the optimal subspace is identified. The matrix exponentiation necessary for the solution of the ODE may also quickly become very expensive, although fast numerical methods have recently been proposed [132].

Fokker-Planck approximation

The firing of reaction j with change vector $\boldsymbol{\nu}_j$ changes the state of the system from $\mathbf{x}$ to $\mathbf{x} - \boldsymbol{\nu}_j$. If the molecular copy number vector is sufficiently large, then the firing of a single reaction does not appreciably change the propensity functions or state probability densities. Hence, the function $P(\mathbf{x} - \boldsymbol{\nu}_j, t)a_j(\mathbf{x} - \boldsymbol{\nu}_j)$, found in the CME (2.56) can be Taylor-expanded using the so-called *Kramer-Moyals expansion*:

$$\begin{aligned} P(\mathbf{x} - \boldsymbol{\nu}_j, t)a_j(\mathbf{x} - \boldsymbol{\nu}_j) = P(\mathbf{x}, t)a_j(\mathbf{x}) - \boldsymbol{\nu}_j^T \frac{\partial}{\partial \mathbf{x}}[P(\mathbf{x}, t)a_j(\mathbf{x})] \\ + \frac{1}{2}\boldsymbol{\nu}_j^T \frac{\partial^2}{\partial \mathbf{x}^2}[P(\mathbf{x}, t)a_j(\mathbf{x})]\boldsymbol{\nu}_j - \dots \end{aligned} \tag{2.59}$$

Substitution of (2.59) into (2.56), and retaining only the first two terms of the expansion gives rise to the so-called **Fokker-Plank** equation:

$$\frac{\partial}{\partial t}P(\mathbf{x}, t) \approx \sum_{j=1}^{R} -\boldsymbol{\nu}_j^T \frac{\partial}{\partial \mathbf{x}}[P(\mathbf{x}, t)a_j(\mathbf{x}, t)] + \frac{1}{2}\boldsymbol{\nu}_j^T \frac{\partial^2}{\partial \mathbf{x}^2}[P(\mathbf{x}, t)a_j(x, t)]\boldsymbol{\nu}_j \tag{2.60}$$

Additionally, if we construct the stoichiometric matrix $\mathbf{S}$ as in (2.49), then (2.59) becomes

$$\frac{\partial}{\partial t}P(\mathbf{x}, t) = -\mathbf{S}^T \frac{\partial}{\partial \mathbf{x}}[P(\mathbf{x}, t)a(\mathbf{x}, t)] + \frac{1}{2}\mathbf{S}^T \frac{\partial^2}{\partial \mathbf{x}^2}[P(\mathbf{x}, t)a(\mathbf{x}, t)]\mathbf{S} \tag{2.61}$$

Thus, under this assumption, the CME (2.56) reduces to a second order partial differential equation. If the system configuration is initially known with certainty, i.e. $P(\mathbf{x}, 0) = \delta(\mathbf{x} - \mathbf{x}_0)$, then (2.61) can be solved analytically to give a Gaussian distribution, see e.g. [167]. We note also that the retention of higher order terms in the expansion

(2.59) has been shown to lead to logical inconsistencies, and thus do not improve the approximation [168].

Langevin approximation

Instead of a partial differential equation for the probability density, the Fokker-Planck equation (2.61) may alternatively be formulated in terms of its equation of motion, as follows. If a stochastic process $\mathbf{X}_t$ evolves according to the CME, then the state of the system at time $t + \Delta t$ is given by $\mathbf{X}_{t+\Delta t} = \mathbf{X}_t + \mathbf{S} \cdot \mathbf{n}$, where $\mathbf{n}$ is a random vector corresponding to number of reaction firings during the interval Δt. If the time interval Δt is sufficiently small such that the propensities are approximately constant over Δt, then each entry n_j of $\mathbf{n}$ is Poisson-distributed (see Section A.4), $n_j \sim \text{Poiss}(a_j \Delta t)$.

The vector $\mathbf{n}$ can be decomposed into a deterministic part $\bar{\mathbf{n}}$ and a stochastic part $\boldsymbol{\xi}$, as $\mathbf{n} = \bar{\mathbf{n}} + \boldsymbol{\xi}$, where $\bar{\mathbf{n}} = \mathbb{E}[\mathbf{n}] = \mathbf{a}\Delta t$ with propensities vector $\mathbf{a} = (a_1(\mathbf{x}), a_2(\mathbf{x}), \ldots)$. Thus

$$\begin{aligned} \mathbf{X}_{t+\Delta t} &= \mathbf{X}_t + \mathbf{S} \cdot (\bar{\mathbf{n}} + \boldsymbol{\xi}) \\ &= \mathbf{X}_t + \mathbf{S} \cdot (\mathbf{a}\Delta t + \boldsymbol{\xi}) \end{aligned} \tag{2.62}$$

where $\mathbb{E}[\boldsymbol{\xi}] = 0$ and $\text{var}[\boldsymbol{\xi}] = \boldsymbol{a}\Delta t$. We then approximate the distribution of the fluctuation $\boldsymbol{\xi}$ by a multivariate Gaussian distribution with mean 0, and covariance $\boldsymbol{a}\Delta t$: $\boldsymbol{\xi} \sim \mathcal{N}(0, \mathbf{a}\Delta t)$. For large $\mathbf{a}\Delta t$, Poisson distributions are well approximated by a normal distribution, motivating this simplification. Finally, we derive

$$\begin{aligned} \mathbf{X}_{t+\Delta t} - \mathbf{X}_t &= \mathbf{S} \cdot \mathbf{a}\Delta t + \mathbf{S} \cdot \xi \\ &= \mathbf{S} \cdot \mathbf{a}\Delta t + \mathbf{S}\sqrt{\text{diag}(\mathbf{a})}\Delta W \end{aligned} \tag{2.63}$$

for a *Wiener process* W, i.e. $\Delta W \sim \mathcal{N}(0, \Delta t)$. In the limit $\Delta t \to 0$ this leads to the so-called **chemical Langevin equation** (CLE):

$$\begin{aligned} d\mathbf{X}_t &= \lim_{\Delta t \to 0} \mathbf{X}_{t+\Delta t} - \mathbf{X}_t \\ &= \mathbf{S} \cdot \mathbf{a}\ dt + \mathbf{S}\sqrt{\text{diag}(\mathbf{a})}dW \end{aligned} \tag{2.64}$$

where $dW = \lim_{\Delta t \to 0} \Delta W$. Eq. (2.64) resembles an ODE with the addition of a stochastic term $\boldsymbol{\xi}$, and is known as a **stochastic differential equation** (SDE). The Wiener process W represents a *Brownian motion*, i.e. the instantaneous displacements are Gaussian distributed, with a variance that grows linearly in time. Using the formulation (2.63), approximate samples from the CME can be generated by sampling a series of Gaussian increments ΔW for a fixed time step of size Δt and updating the state $X_t \to X_{t+\Delta t}$ accordingly, a procedure known as Euler-Maruyama integration of the SDE [169].

System size expansion

The following subsection details the **system size expansion** of the chemical master equation, as introduced by Van Kampen [167]. This expansion has been studied extensively since its inception, including several recent papers by Grima and coworkers, see [170].

Let $\mathbf{n} \in \mathbb{N}_0^N$ denote the vector of molecular copy numbers, with probability density $P(\mathbf{n}, t)$ at time t, and let Ω denote the volume of the reaction system, e.g. the cellular

volume if the CRN describes intracellular chemical reactions. The concentration is given by the ratio $\boldsymbol{\phi} = \mathbf{n}/\Omega$, see Section 2.4.1.

The canonical "linear noise" ansatz assumes that the fluctuations of the state of the system about the macroscopic mean are small, of the order $\sqrt{\Omega}$, such that the state of the system is given by:

$$\mathbf{n} = \boldsymbol{\phi}\Omega + \boldsymbol{\eta}\Omega^{\frac{1}{2}} \tag{2.65}$$

where the fluctuation $\boldsymbol{\eta} \in \mathbb{R}^N$, and $\boldsymbol{\phi} \in \mathbb{R}^N_{0,+}$ is the macroscopic mean concentration, i.e. in the limit of vanishing stochastic fluctuations. The macroscopic mean for each species $i = 1, \ldots, N$ evolves according to the MRE, see Section 2.4.1:

$$\frac{d\boldsymbol{\phi}_i}{dt} = \sum_{j=1}^{R} \mathbf{S}_{ij} \cdot f_j(\boldsymbol{\phi}, t, \boldsymbol{\Theta}) \tag{2.66}$$

where R is the number of reactions, $\mathbf{S}$ is the stoichiometric matrix of the chemical reaction network and $\mathbf{f}$ is the vector of reaction fluxes. This can also be simply written as $\frac{d\boldsymbol{\phi}}{dt} = \mathbf{S} \cdot \mathbf{f}(\boldsymbol{\phi}, t, \boldsymbol{\Theta})$, where $\mathbf{f}(\boldsymbol{\phi}, t, \boldsymbol{\Theta})$ depends on the macroscopic means $\boldsymbol{\phi}$ and the reaction constants $\boldsymbol{\Theta}$. Note that the MRE is identical to the deterministic part of the CLE (2.64), with $\mathbf{f} = \mathbf{a}$.

The term $\boldsymbol{\eta}$ plays the role of the stochastic fluctuation about the mean. Using the transformation (2.65), the CME (2.56) can be written in terms of a new probability density in $\boldsymbol{\eta}$, $\Pi(\boldsymbol{\eta}, t)$, since the stochasticity of the system is entirely due to the fluctuation:

$$\begin{aligned} \frac{\partial P(\mathbf{n}, t)}{\partial t} &= \frac{\partial \Pi(\boldsymbol{\eta}, t)}{\partial t} + \sum_{i=1}^{N} \frac{\partial \Pi(\boldsymbol{\eta}, t)}{\partial \eta_i} \frac{\partial \eta_i}{\partial t} \\ &= \frac{\partial \Pi(\boldsymbol{\eta}, t)}{\partial t} - \Omega^{-1/2} \sum_{i=1}^{N} \frac{\partial \Pi(\boldsymbol{\eta}, t)}{\partial \eta_i} \frac{\partial \phi_i}{\partial t} \end{aligned} \tag{2.67}$$

utilizing $\frac{\partial \boldsymbol{\eta}}{\partial t} = -\Omega^{-\frac{1}{2}} \frac{\partial \boldsymbol{\phi}}{\partial t}$ which follows from differentiating the ansatz (2.65), assuming constant $\mathbf{n}$.

We next introduce the step operator $\mathbb{E}_k^{\Delta}$ ($\Delta \in \mathbb{Z}$), which acts on a function $g(\mathbf{n})$ by incrementing the value of n_k by Δ: $\mathbb{E}_k^{\Delta} g(\mathbf{n}) = g(n_1, \ldots, n_k + \Delta, \ldots, n_N)$. Using the linear noise ansatz (2.65), we see that the step operator $\mathbb{E}_k^{\Delta}$ acts on a function of the stochastic fluctuation $\boldsymbol{\eta}$ by raising η_k by $\Omega^{-1/2}\Delta$. Hence, repeated application of the step operator with $\Delta = \mathbf{S}_{ij}$ to a function $g(\boldsymbol{\eta})$ yields:

$$\begin{aligned} \prod_{i=1}^{N} E_i^{-\mathbf{S}_{ij}} g(\boldsymbol{\eta}) &= g(\eta_1 - \Omega^{-1/2}\mathbf{S}_{1j}, \eta_2 - \Omega^{-1/2}\mathbf{S}_{2j}, \ldots, \eta_N - \Omega^{-1/2}\mathbf{S}_{Nj}) \\ &= g(\boldsymbol{\eta} - \Omega^{1/2}\boldsymbol{\nu}_j) \end{aligned} \tag{2.68}$$

where $\boldsymbol{\nu}_j = (\mathbf{S}_{1j}, \mathbf{S}_{2j}, \ldots, \mathbf{S}_{Nj})^T$.

The CME (2.56) can then easily be rewritten in terms of the step operator as follows:

$$\begin{aligned}\frac{\partial}{\partial t}P(\mathbf{n},t) &= \Omega\sum_{j=1}^{R} P(\mathbf{n}-\boldsymbol{\nu}_j,t)a_j(\mathbf{n}-\boldsymbol{\nu}_j) - P(\mathbf{n},t)a_j(\mathbf{n}) \\ &= \Omega\sum_{j=1}^{R}\prod_{i=1}^{N}\mathbb{E}_i^{-\mathbf{S}_{ij}}\left[P(\mathbf{n},t)a_j(\mathbf{n})\right] - P(\mathbf{n},t)a_j(\mathbf{n}) \\ &= \Omega\sum_{j=1}^{R}\left(\prod_{i=1}^{N}\mathbb{E}_i^{-\mathbf{S}_{ij}} - 1\right)P(\mathbf{n},t)a_j(\mathbf{n})\end{aligned} \tag{2.69}$$

The prefactor Ω is a matter of convention, and can also be absorbed into the propensity functions $a_j(\mathbf{n})$. By computing the action of the compound step operator (2.68) on the test function $g(\mathbf{n})$ and Taylor-expanding about $\boldsymbol{\eta}$, we observe that $\mathbb{E}_i^{-\mathbf{S}_{ij}}$ can be recast as a differential operator in terms of $\boldsymbol{\eta}$:

$$\begin{aligned}\left(\prod_{i=1}^{N}\mathbb{E}_i^{-\mathbf{S}_{ij}} - 1\right)g(\boldsymbol{\eta}) &= g(\boldsymbol{\eta}-\Omega^{-1/2}\boldsymbol{\nu}_j) - g(\boldsymbol{\eta}) \\ &= g(\boldsymbol{\eta}) - \Omega^{-1/2}\boldsymbol{\nu}_j^T\frac{\partial}{\partial\boldsymbol{\eta}}g(\boldsymbol{\eta}) + \ldots - g(\boldsymbol{\eta}) \\ \Rightarrow \qquad \prod_{i=1}^{N}\mathbb{E}_i^{-\mathbf{S}_{ij}} - 1 &= -\Omega^{-1/2}\boldsymbol{\nu}_j^T\frac{\partial}{\partial\boldsymbol{\eta}} + \frac{\Omega^{-1}}{2}\boldsymbol{\nu}_j^T\frac{\partial^2}{\partial\boldsymbol{\eta}^2}\boldsymbol{\nu}_j \\ &\quad - \frac{\Omega^{-3/2}}{6}\sum_{i,k,r=1}^{N}\mathbf{S}_{ij}\mathbf{S}_{kj}\mathbf{S}_{rj}\frac{\partial^3}{\partial\eta_i\partial\eta_k\partial\eta_r} + \mathcal{O}(\Omega^{-2})\end{aligned} \tag{2.70}$$

Next, we similarly expand each rate function $a_j(\mathbf{n})$ about the macroscopic mean $\boldsymbol{\phi}$ using (2.65):

$$\begin{aligned}a_j &= f_j(\boldsymbol{\phi},t,\boldsymbol{\Theta}) + \Omega^{-1/2}\sum_{w=1}^{N}\eta_w\frac{\partial f_j(\boldsymbol{\phi},t,\boldsymbol{\Theta})}{\partial\phi_w} \\ &\quad + \frac{\Omega^{-1}}{2}\left(\sum_{w,z=1}^{N}\eta_w\eta_z\frac{\partial^2 f_j(\boldsymbol{\phi},t,\boldsymbol{\Theta})}{\partial\phi_w\partial\phi_z} - \sum_{w=1}^{N}\frac{\partial^2 f_j(\boldsymbol{\phi},t,\boldsymbol{\Theta})}{\partial\phi_w^2}\right) \\ &\quad + \mathcal{O}(\Omega^{-3/2})\end{aligned} \tag{2.71}$$

which holds for all elementary chemical reactions involving at most two molecules, see [170] and Section 2.4.1 for details. Here $f_j(\boldsymbol{\phi},t,\boldsymbol{\Theta})$ is the macroscopic rate equation as in (2.66).

Finally, by combining the expression for the CME in operator form (2.69), the Taylor expansion of the step operator in inverse powers of $\Omega^{1/2}$ (2.70), and the expansion of the microscopic rate equations in terms of the macroscopic concentration $\boldsymbol{\phi}$ (2.71), we achieve

the partial differential equation for the time evolution of the probability density of the stochastic fluctuation $\boldsymbol{\eta}$:

$$\begin{aligned}\frac{\partial \Pi(\boldsymbol{\eta}, t, \boldsymbol{\Theta})}{\partial t} = &- \sum_{i,w=1}^{N} \mathbf{J}_{iw} \frac{\partial}{\partial \eta_i}(\eta_w \Pi) + \frac{1}{2} \sum_{i,k=1}^{N} \mathbf{D}_{ik} \frac{\partial^2}{\partial \eta_i \partial \eta_l} \Pi \\ &- \frac{\Omega^{-1/2}}{2} \left(\sum_{i,w,z=1}^{N} \frac{\partial \mathbf{J}_{iw}}{\partial \phi_z} \frac{\partial}{\partial \eta_i}(\eta_w \eta_i \Pi) - \sum_{i,w=1}^{N} \phi_w \frac{\partial \mathbf{J}_{iw}}{\partial \phi_w} \frac{\partial}{\partial \eta_i} \Pi \right) \\ &- \frac{\Omega^{-1/2}}{6} \sum_{i,k,r=1}^{N} \mathbf{D}_{ikr} \frac{\partial^3}{\partial \eta_i \partial \eta_k \partial \eta_r} \Pi + \mathcal{O}(\Omega^{-1})\end{aligned} \tag{2.72}$$

with

$$\mathbf{J}_{iw} = \sum_{j=1}^{R} \mathbf{S}_{ij} \frac{\partial f_j(\boldsymbol{\phi}, t, \boldsymbol{\Theta})}{\partial \phi_w} \tag{2.73}$$

and

$$\mathbf{D}_{ik\ldots r} = \sum_{j=1}^{R} \mathbf{S}_{ij} \mathbf{S}_{kj} \ldots \mathbf{S}_{rj} f_j(\boldsymbol{\phi}, t, \boldsymbol{\Theta}). \tag{2.74}$$

The matrix $\mathbf{J}$ is the Jacobian of the macroscopic rate equation, and $\mathbf{D}$ is a generalized diffusion matrix.

The result of this rather complicated derivation is an expression for the evolution of the probability density $\Pi(\boldsymbol{\eta}, t)$ of the stochastic fluctuation $\boldsymbol{\eta}$. Keeping only the zeroth-order term in $\Omega^{\frac{1}{2}}$ leads to the so-called "linear noise approximation" (LNA), as first introduced by Van Kampen [167]:

$$\frac{\partial \Pi(\boldsymbol{\eta}, t, \boldsymbol{\Theta})}{\partial t} = - \sum_{i,w=1}^{N} \mathbf{J}_{iw} \frac{\partial}{\partial \eta_i}(\eta_w \Pi) + \frac{1}{2} \sum_{i,k=1}^{N} \mathbf{D}_{ik} \frac{\partial^2}{\partial \eta_i \partial \eta_k} \Pi \tag{2.75}$$

The LNA has the form of a linear Fokker-Planck equation, see Section 2.4.2. Computing the expectation (denoted using angular brackets) of the stochastic fluctuation, $\langle \boldsymbol{\eta} \rangle = \int \boldsymbol{\eta} \Pi(\boldsymbol{\eta}, t, \boldsymbol{\Theta}) d\boldsymbol{\eta}$, gives

$$\frac{\partial \langle \boldsymbol{\eta} \rangle}{\partial t} = \mathbf{J} \cdot \langle \boldsymbol{\eta} \rangle. \tag{2.76}$$

The second moment of the stochastic fluctuation $\langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle$, evolves as:

$$\frac{\partial \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle}{\partial t} = \mathbf{J} \cdot \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle + \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle \cdot \mathbf{J}^T + \mathbf{D} \tag{2.77}$$

where $\mathbf{D} = \mathbf{S} f(\boldsymbol{\phi}, t, \boldsymbol{\Theta}) \mathbf{S}^T$. Hence the covariance matrix $\mathbf{V}(t) = \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle - \langle \boldsymbol{\eta} \rangle \langle \boldsymbol{\eta}^T \rangle$ has derivative

$$\begin{aligned}\frac{\partial}{\partial t} \mathbf{V}(\boldsymbol{\eta}(t)) &= \mathbf{J} \cdot \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle + \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle \cdot \mathbf{J}^T + \mathbf{D} - \left(\frac{\partial \langle \boldsymbol{\eta} \rangle}{\partial t} \langle \boldsymbol{\eta}^T \rangle + \langle \boldsymbol{\eta} \rangle \frac{\partial \langle \boldsymbol{\eta}^T \rangle}{\partial t} \right) \\ &= \mathbf{J} \cdot \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle + \langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle \cdot \mathbf{J}^T + \mathbf{D} - \left(\mathbf{J} \cdot \langle \boldsymbol{\eta} \rangle \langle \boldsymbol{\eta}^T \rangle + \langle \boldsymbol{\eta} \rangle \langle \boldsymbol{\eta}^T \rangle \cdot \mathbf{J}^T \right) \\ &= \mathbf{J} \cdot \left[\langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle - \langle \boldsymbol{\eta} \rangle \langle \boldsymbol{\eta}^T \rangle \right] + \left[\langle \boldsymbol{\eta} \boldsymbol{\eta}^T \rangle - \langle \boldsymbol{\eta} \rangle \langle \boldsymbol{\eta}^T \rangle \right] \cdot \mathbf{J}^T + \mathbf{D} \\ &= \mathbf{J}\mathbf{V} + \mathbf{V}\mathbf{J}^T + \mathbf{D}\end{aligned} \tag{2.78}$$

Taken together, (2.76) and (2.78) yield a set of ODEs for the mean and covariance of the stochastic fluctuations, from which the probability distribution can be recovered.

Finally, if one retains the next higher order terms in the expansion (2.72) of order $\Omega^{-1/2}$, one may compute more accurate expressions for the evolution of the first two moments of the stochastic fluctuations, $\langle \boldsymbol{\eta} \rangle$ and $\langle \boldsymbol{\eta}\boldsymbol{\eta}^T \rangle$. These expressions constitute the so-called **empirical mesoscopic rate equations,** (EMRE) and are generally more accurate than the LNA since they include higher order corrections; for example EMREs were recently used to discover the existence of a novel inversion effect that under certain circumstances can reverses the relative order of concentrations of chemical species when applying the macroscopic rate equations to systems with subcritical reaction volumes [94]. Details for computing the EMRE are found in [170]. We also note that the LNA (and SSE in general) provides an accurate approximation to the true probability distribution only for systems which possess a unique, globally stable macroscopic solution, i.e. for monostable systems, see e.g. [167]. For systems possessing locally stable solutions other approaches must be employed.

2.4.3 Stochastic simulation

The CME is often analytically intractable due to its infinite dimensionality. Luckily, in addition to various approximations, a number of numerical methods have also been developed for exact simulation of the chemical master equation. Essentially, using the propensities defined in Table 2.1, one may draw exact samples from the underlying Markov jump process described by the CME. The **stochastic simulation algorithm** (SSA) is a procedure for simulating trajectories such that each realization occurs with frequency proportional to the probability of that trajectory as given by the CME.

The SSA (a.k.a. the Gillespie Algorithm) [117] is conceptually quite simple. It was conceived using a simple physical argument, assuming a well-mixed reaction environment, and thermal equilibrium; it was later rederived in a more rigorous manner [96]. The starting point is a chemical reaction network with R possible reactions, and the corresponding set of reaction constants. The propensities are then computed for each reaction, see Table 2.1 and Section 2.4.1, with reactions involving species with insufficient molecules for the reaction to take place having zero propensity. So long as the configuration of the system (i.e. the number of molecules of each species) does not change, the propensities remain constant. Hence, the waiting time τ_j until the j^{th} reaction occurs is exponentially distributed with parameter equal to the reaction propensity: $\tau_j \sim \text{Exp}(a_j(\mathbf{X}))$.

The individual reaction firings are exponentially distributed, and thus collectively form a system of competing exponential processes, see Section A.3. Hence, it is easy to compute the probability that reaction k will be the next reaction to fire: the probability of reaction k with propensity $a_k(\mathbf{X})$ occurring next is simply $P_k(\mathbf{X}) = a_k(\mathbf{X})/a_0(\mathbf{X})$. with $a_0(\mathbf{X}) = \sum_{k=1}^{R} a_i(\mathbf{X})$. Of course the probabilities and propensities are functions of the instantaneous configuration of the system, and the kinetic constants of the CRN.

The time of the next reaction is also determined as in Eq. (A.7). The waiting time until each event is exponentially distributed, and the waiting time until the first event is distributed as $\tau \sim \text{Exp}(a_0(\mathbf{X}))$. Some computational effort can be avoided by computing only the waiting time until the first reaction occurs, and thus avoiding further sampling

of exponential random variables. The variant of the SSA implementing this strategy is known as the "next reaction" SSA [171].

The SSA described above generates exact samples from the CME, and thus has been a tremendous asset to the chemical kinetic community. However, the SSA is often very computationally intensive. For instance, in the case of fast isomerization reactions where species interconvert, millions of reactions may take place in a short amount of time, before any "interesting" dynamics take place, involving slower (i.e. lower propensity) reactions. Thus, SSA may be impractical for the study of dynamics of systems involving disparate time scales. However, many variations of SSA have been developed since the 70's for accelerating the SSA, see e.g. [118–122, 124, 128, 172–174] .

Part II

Results

Chapter 3

Multiresolution Correlation Analysis

In the following chapter, I present a new method for investigating the local correlation structure of low-dimensional datasets, e.g. as may arise during qPCR or image-based quantification of transcription factors using immunohistological staining. The tool has been published in the following original research article:

> **Feigelman, J.,** Theis, F. J., & Marr, C. (2014). MCA: Multiresolution Correlation Analysis, a graphical tool for subpopulation identification in single-cell gene expression data. BMC Bioinformatics, 15(1), 1-10.

The method was initially developed as a graphical tool to address the question of robustness of inferred partial correlation networks arising in the analysis of mouse embryonic stem cell colonies. In particular, we were interested in understanding whether the expression level of Nanog played a role in the network, and whether subpopulations that differ in the expression of one of several pluripotency factors showed differential correlation networks. During the development of this tool, it became clear that it was more versatile, affording insight into the structure of the correlation networks, stability of inferred correlations, detection of novel subpopulations, and identification of potentially outliers that can skew correlation estimates.

We present the method and the application to previously published mESC data. In Chaper 6, we apply the method to novel mESC data sets.

3.1 Introduction

Heterogeneity in cellular populations has been the focus of many recent publications in areas such as embryonic stem cells [24], induced pluripotency [175], transcriptomics [176], and metabolomics [177]. In biological experiments, data often originate from a mixture of qualitatively differing subpopulations corresponding to e.g. distinct phenotypes in assays of cellular populations. [178]. For example, whole blood samples contain a mixture of distinct cell lineages which can be identified based on the presence of lineage-specific cell surface markers [179]. Embryonic stem cells have also been shown to exhibit heterogeneous

expression of pluripotency factors critical for the maintenance of pluripotency in culture [24][39]. Indeed, there is increasing evidence for the existence of cellular subpopulations with possible noise-induced transitions between phenotypic attractors [180]. Thus it is clear that traditional techniques, which provide only population averages, may fail to resolve the true population heterogeneity.

Technologies such as flow cytometry, single-cell qPCR, mass cytometry and time-lapse fluorescent microscopy are uniquely positioned to answer questions regarding the makeup of cellular populations. Each is able to yield quantitative measurements of cellular state, i.e. mRNA expression or protein copy number, which may be representative of the underlying subpopulations.

If the subpopulations are not already known, various methods exist to attempt to learn them on the basis of the data distribution. Classical techniques such as clustering may be useful for subpopulation identification if the subpopulations are readily separable in terms of expression levels [181]. Alternatively, more sophisticated machine-learning based approaches such as mixture models, (fuzzy) k-means clustering, multilayer perceptrons, self organizing maps, support vector machines, regression trees, and many others have also been applied to subpopulation identification (see Lugli *et al.* [54] and Bashashati *et al.* [182] for a review of subpopulation identification approaches applied to flow cytometry).

However, existing methods for subpopulation identification predominantly rely on heterogeneous expression levels. If the distributions overlap, identification of individual subpopulations based on expression alone may be difficult. In the case where subpopulations exhibit differential regulation motifs, they may be identifiable based on their distinctive correlations. Examining the local, state-dependent correlation of covariates provides additional information regarding the underlying distributions attributable to distinct subpopulations. In particular, we expect correlations to change for regions of state space (i.e. the space of possible gene expression levels) containing predominantly samples from a single subpopulation. Correlation analysis in subspaces of high dimensional data have gained attention over the past several years, particularly in the context of data mining e.g. in databases. For instance, algorithms such as MAFIA [183], CURLER [184], δ-Clusters [185] , ENCLUS [186], etc. have been proposed for automatic identification of clusters using lower-dimensional subspaces. However, automatically identified clusters may be difficult to interpret biologically, and it may be difficult to assess their relative robustness.

We introduce a complementary method, Multiresolution Correlation Analysis (MCA) for systematically examining the dependence of local correlation upon location in state space. Using MCA, the correlations of pairs of variables are examined for regions of state space subdivided with varying granularity. The analysis can be summarized using MCA plots, which provide a visual representation of the pairwise correlation as a function of expression of a third variable.

MCA plots simultaneously visualize the correlations of data subsets of all sizes, centered at all locations in the distribution of a sorting variable, making it possible to distinguish regions with robust correlations which may be indicative of distinct subpopulations. Lastly, they provide the ability to identify observations which contribute disproportionately to the overall correlation structure, and hence skew the estimated correlation of the entire population.

3.2 Results

3.2.1 MCA reveals differential regulation of subpopulations in simulated gene expression data

To evaluate the MCA approach, we simulated gene expression data using a simple three species gene regulatory motif, given by Equation (3.5) as described in *Methods*. In this system, Z activates X and X activates Y (Figure 3.1A, left) via Hill-type activation functions, and population-level heterogeneity is introduced via the use of stochastic differential equations which approximate the intrinsic noisiness of gene expression [187][188][29].

The steady state distribution resulting from a typical simulation (Figure 3.1A, center) shows a significant positive Pearson correlation ($p < 0.05$) between Z and X, and between X and Y (Figure 3.1A, right), and no significant correlation between Z and Y, as would be expected from the underlying regulatory motif.

Similarly, we simulated a biological system for which Z activates X, but where X inhibits Y (see Figure 3.1B, left) and Equation (3.6) of *Methods*). The resulting steady state distribution (Figure 3.1B, center) appears similar to that of the activation model. However, correlation analysis reveals that Z and X show significant positive correlation, and X and Y significant negative correlation (Figure 3.1B, right), in accordance with the underlying biological motif. The Pearson correlation also indicates significant negative correlation between Y and Z in the inhibition model, an indirect effect.

When combining the steady state distributions from activation and inhibition models (Figure 3.1C), the net Pearson correlation between X and Y is significantly negative (Figure 3.1C, I). Absent of subpopulation analysis, we would conclude that the relationship between expression levels of X and Y is antagonistic, implying an inhibitory motif.

In contrast, performing the same analysis on the subpopulation with Z expression levels in the lowest 30% of the Z-distribution (Figure 3.1C, II) yields a significant positive correlation between X and Y. Likewise, performing correlation analysis on the samples in the top 30% of the Z-distribution shows just the opposite, a significant negative correlation between X and Y (Figure 3.1C, III).

We can combine all of the Z-sorted subpopulations of varying size together using the MCA plot (Figure 3.1D), constructed as described in *Methods*. Briefly, the MCA plot shows the correlation of a pair of factors, for subpopulations defined by a sorting variable. The abscissa indicates the median value of the sorting variable for that subpopulation and the ordinate indicates the fraction of the population included in that subpopulation. Thus, higher points indicate larger subpopulations, points to the left indicate lower overall expression of the sorting variable, points to the right higher overall expression, etc. The regions where the computed correlation is statistically significant ($p < 0.05$) are indicated.

By systematic inspection via the MCA plot, we can conclude that subpopulations with low Z values indeed show significant positive correlation between X and Y (Figure 3.1D, blue region), and subpopulations with high Z values show significant negative correlation between X and Y (Figure 3.1D, red region, see *Methods* for details).

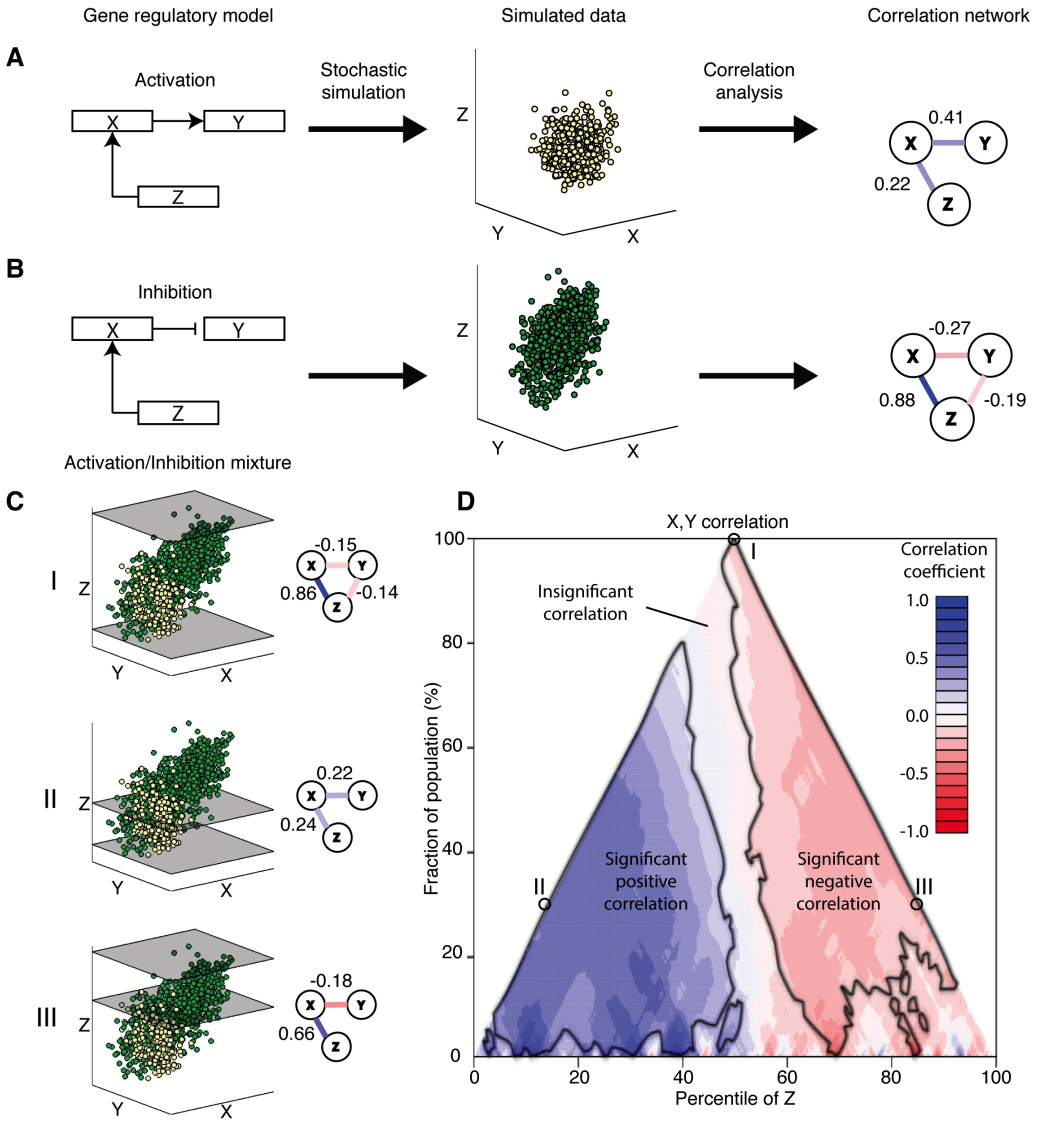

Figure 3.1: MCA reveals the presence of subpopulations with differential regulation. **A.** A three species activation motif (left), its steady state distribution (center) from an SDE simulation, and the resultant correlation network (right), showing positive correlation for species with an activating interaction. **B** A three species activation/inhibition motif induces positive correlation corresponding to activation and negative correlation corresponding to inhibition. **C. I.** Mixture of the activation and inhibition steady state data depicted in A and B. **II.** Correlation analysis of the subset from the lowest 30% of the Z-distribution shows significant positive X, Y correlation. **III.** Correlation analysis of the subset from the highest 30% of the Z-distribution shows significant negative X, Y correlation. **D.** Combining all subpopulations sorted by median Z value and subpopulation size into an MCA plot reveals robust separation of positive and negative correlations for subpopulations with low or high Z values, respectively.

3.2.2 MCA plots as a diagnostic tool for transcriptomic analysis

MCA plots can be used to provide a multiresolution view of the correlation structure of real transcriptomic data. This allows us to confirm previous conclusions regarding heterogeneous subpopulations, detect potential novel subpopulations, and provides insight into the origin of the observed correlations.

We used MCA to analyze previously published single-cell transcriptomic data obtained from mouse embryonic stem cells (mESCs) [40][32]. There, microfluidic single-cell qPCR was used to obtain the relative expression of mRNAs for eight transcription factors known to be involved in regulation of pluripotency in mESCs: Fgf5, Nanog, Oct4, Sox2, Rex1, Pecam1, Stella and Gbx2, and Gapdh, a housekeeping gene against which all other transcript copy numbers were normalized. Analysis of subpopulations showed difference in the correlation networks of Nanog+/- and Fgf5+/- subpopulations, as well as clear separation of subpopulations using principal component analysis.

After data cleaning and normalization according to the method of Trott *et al.* [40], we generated the MCA plots for all pairs of genes, for all possible sortings, using Pearson correlation and a significance cutoff of $p < 0.05$. All points with $p > 0.05$ are colored white in the MCA plot.

Detection of robust correlations

In an MCA plot, correlations that are globally robust with respect to changes in the sorting variable are easily distinguished by uniform coloration. For example, the correlation of Rex1 and Sox2 is robust with respect to changes in Pecam1 expression (Figure 3.2A, top). The scatter plot of Rex1 and Sox2 is shown for reference (Figure 3.2A, bottom). The robust positive correlation of Rex1 and Sox2 is consistent with current models of transactivation of Sox2 by Rex1 [189].

Outlier detection

Correlation analysis can be sensitive to one or a few samples which substantially alter the estimated correlation of the entire population. In such a case, all subpopulations including these samples show a significant correlation, whereas their exclusion results in no significant correlation or potentially correlation of the opposite sign. MCA plots are able to detect such samples and identify them as sources of the detected correlation. For example, when sorting by Sox2, all subpopulations which do not contain the sample with the highest Sox2 expression do not show statistically significant correlation between Rex1 and Gbx2, whereas all subpopulations that do include this point show significant positive correlation (Figure 3.2B, top). Upon inspection of the data (Figure 3.2B, bottom) it is obvious that this single point, indicated by the arrow, is an outlier. Exclusion of this point renders the Rex1, Gbx2 correlation insignificant.

Subpopulation identification

MCA plots are useful for identification of interesting subpopulations as shown for synthetic data (Figure 3.1C). Regions exhibiting a robust correlation may indicate the presence of differential regulation or a distinct cellular phenotype. For instance, sorting by Stella

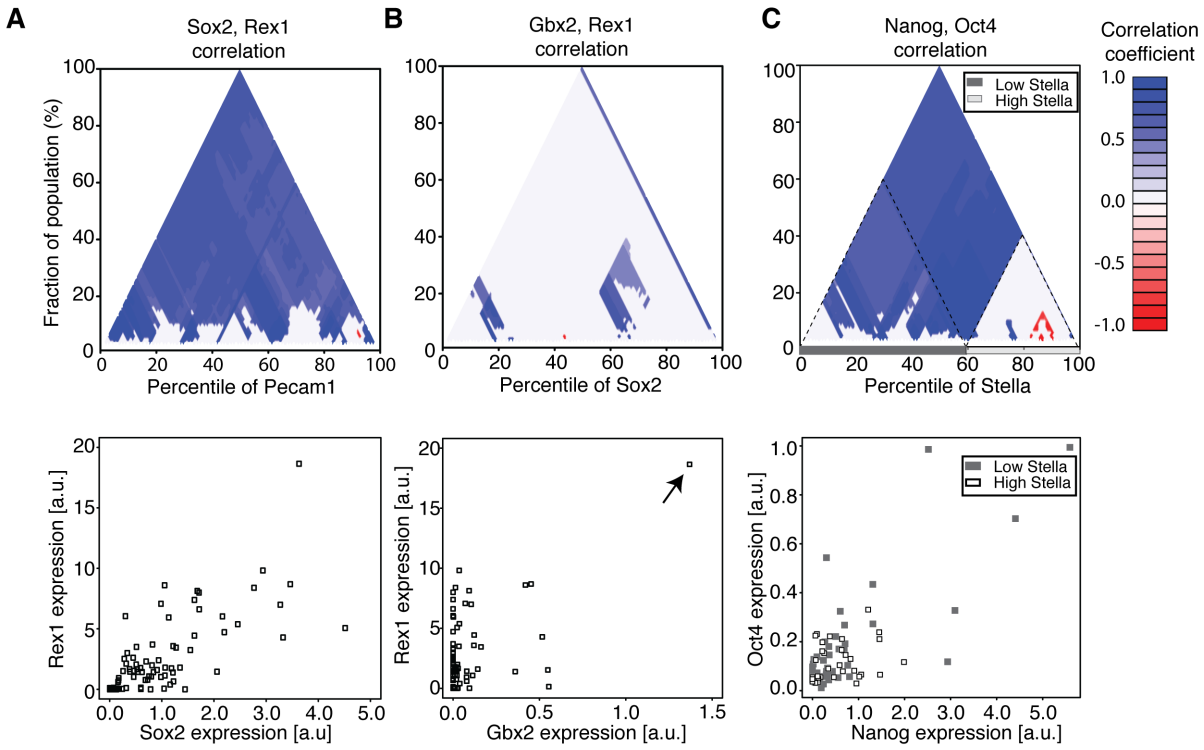

Figure 3.2: MCA plots reveal important features of the correlation structure in single-cell transcriptomics data. **A.** MCA plots with uniform appearance (top) reveal robust correlations amongst pairs of variables (scatter plot, bottom) like Rex1 and Sox2, sorted by Pecam1. **B.** Outliers can easily be detected via characteristic diagonal stripe patterns. Here a single sample with the highest value in the Sox2 distribution is enough to induce an overall positive Gbx2, Rex1 correlation (bottom, arrow). **C.** Robust subpopulations can be identified. The presence of a large triangular region with uniform correlation or lack of correlation between Rex1 and Nanog may indicate a subpopulation, seen here for cells from the highest 40% of the Stella distribution (top). The cells from the high Stella compartment (open boxes) are not significantly correlated for Rex1 and Nanog, in contrast to those from the low Stella compartment (filled boxes, bottom).

reveals the presence of a large region (the highest 40% of the population) for which the correlation between Nanog and Oct4 is not statistically significant (Figure 3.2C, top). Conversely, including the cells from the lowest 60% of the Stella distribution is sufficient to induce a significant positive correlation (Figure 3.2C, top). Inspection of the scatter plot of Nanog and Oct4 (Figure 3.2C, bottom) confirms that the lower 60% is noticeably more correlated than the top 40%. Hayashi *et al.* [32] note that mESCs with low or absent Stella expression may be more representative of epiblast-derived stem cells, and thus are expected to show differential regulation from the high Stella cells, which are more embryonic stem cell-like. Interestingly, the possibility of antagonistic regulation between Oct4 and Nanog in mESCs has recently also been raised [190].

3.2.3 MCA provides additional insight into previously described subpopulations

In order to identify subpopulations with different co-expression networks, Trott *et al.* [40] grouped cells according to normalized pluripotency gene expression. Networks are constructed on the basis of significant Pearson correlation between nodes, and subdivided into groups based on the presence of two heterogeneously expressed transcription factors, Nanog and Fgf5. The high Nanog (Nanog+) compartment was defined such that Fgf5 expression is absent for all cells with Nanog expression at or above the minimum level of this compartment.

MCA plots confirm differential Gbx2, Sox2 correlation for high Nanog cells

As in their study, we find that the Nanog+ subpopulation indeed has a significant positive Pearson correlation between Gbx2 and Sox2 (Figure 3.3A, I). Also in agreement, the remaining cells (Nanog-, $0^{th} - 74^{th}$ percentile), show no significant correlation between Gbx2 and Sox2 (Figure 3.3A, II). However, we learn from the MCA plot that in fact only the top 10% contribute to the observed positive correlation; the subset of the high Nanog subpopulation between the 74^{th} and 93^{rd} percentile (Figure 3.3A, III) is not significantly correlated ($p = 0.57$).

MCA plots show that Gbx2, Sox2 correlations are not robust for Fgf5- cells

The authors found that the 15 of 83 cells (18%) expressing Fgf5 (Fgf5+ compartment) do not correlate for Gbx2 and Sox2, whereas the remaining 68 Fgf5- cells (82%) show a significant positive correlation [40] . Using an MCA plot we see that this indeed true (Figure 3.3B, I and II for Fgf5+, Fgf5-, respectively). However it is also evident that the Fgf5+ cells with Fgf5 expression between the 90^{th} and 100^{th} percentile of the distribution are in fact positively correlated for Gbx2 and Sox2 (Figure 3.3B, III). Likewise, the majority of the cells in the Fgf5- compartment are not significantly correlated for Gbx2 and Sox2. Indeed most subpopulations consisting of cells with expression between the 0^{th} and 75^{th} percentile of the Fgf5 distribution are not significantly correlated for Gbx2 and Sox2 ($p > 0.05$). Thus, MCA provides the means for a detailed and robust subpopulation identification, superior to *ad hoc* compartmentalization.

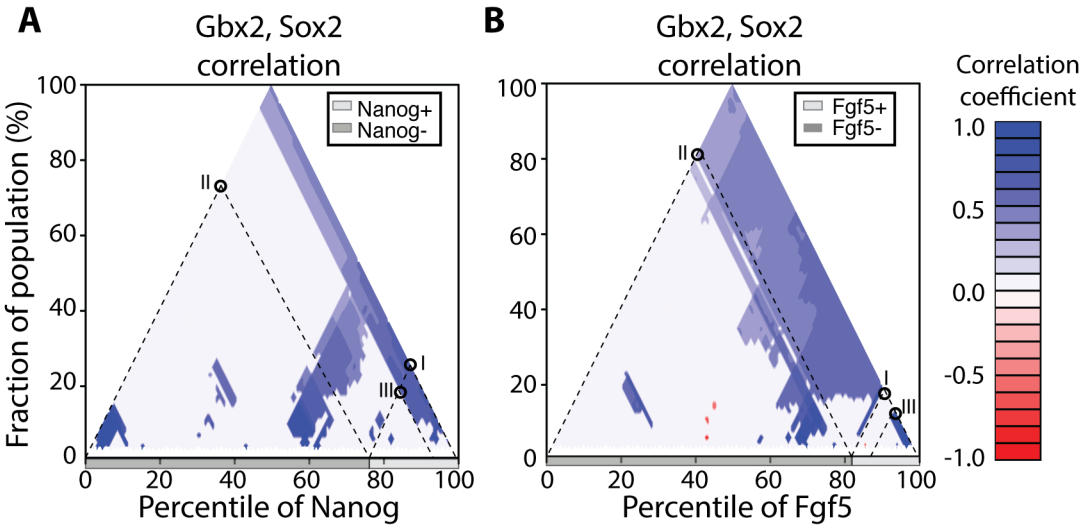

Figure 3.3: MCA plots identify interesting biological subpopulations in mouse embryonic stem cells. **A.** MCA analysis reveals insight into the influence of Nanog on the Gbx2, Sox2 interaction. Gbx2 and Sox2 are significantly positively correlated when considering the entire Nanog+ compartment (quantiles 74% to 100% of Nanog, **I**). When considering the remaining Nanog- cells, the correlation is no longer significant (quantiles 0% to 74%, **II**). MCA plots reveal that the positive correlation in the Nanog+ compartment is due to just half of the compartment; the rest is uncorrelated (**III**). **B.** Gbx2 and Sox2 are uncorrelated when considering the whole Fgf5+ compartment (quantiles 82% to 100%, **I**). However, the top 10% are significantly positively correlated when considered alone (**II**). The Fgf5- compartment is significantly positively correlated (**III**), however, the majority of subpopulations in the Fgf5- compartment are not significantly correlated. See main text for a comparison of our findings with the previous report of Trott *et al.* [40].

3.3 Discussion

Fueled by newly developed single-cell technologies such as single-cell transcriptomic [84][191], genomic [83] and proteomic [56] analysis, many new methods have emerged which attempt to shed light on cellular heterogeneity [192][51][193][194].

Previous methods for the detection of heterogeneous subpopulations in biological data have largely focused on grouping observations according to expression level, and thus requires that subpopulations be readily separable. For instance, in FACS cellular subpopulations are often identified with manually determined compartments [195][196][197]. If the data are easily separated, clustering methods such as Gaussian mixture modeling and k-means clustering have proved well suited to this task [181].

Alternatively, methods such as principal component analysis attempts to identify the principal directions, along which the data are maximally separated [198]. Data which cluster together in the reduced dimensional subspace spanned by the first few principal components are thought to be representative of subpopulations. A similar method was employed by Trott *et al.* when analyzing the Fgf5+/- and Nanog +/- compartments [40]. Non-linear alternatives to PCA including Gaussian Process Latent Variable Modeling have also recently been shown to be useful for the identification of cellular subpopulations [194][60].

None of the previously mentioned methods utilize correlation information in the identification of cellular subpopulations, with the exception of Gaussian mixture modeling which attempts to learn the correlation matrices of Gaussian distributions thought to have generated the data. However, as shown here, the local correlation structure provides additional insight into the existence of differentially regulated subpopulations and hence should not be disregarded.

To date, relatively few methods have addressed the possibility of local, state-dependent correlations. Chen *et al.* [199] developed a method for analyzing the effect of local non-linear correlations in gene expression data, and applied it to a microarray dataset; a similar method was recently developed by Tjøstheim *et al.* [200] for estimating local Gaussian correlation in the context of econometric data. However, these methods required the definition of a interaction scale for the computation of local correlations or consider only the relative distance between data points and not their absolute levels when computing local correlations.

Recently Cordeiro *et al.* [201], developed a sophisticated algorithm for identifying clusters of arbitrary orientation, also in a multiresolution context. MCA is not as general in that it does not consider clusters aligned along arbitrary projections of the data but provides instead a comprehensive, multiresolution view of the correlation structure according to the measured covariates, preserving expression-level dependencies while not requiring any predefined bandwidth or interaction distance, and thus may provide more biological insight into the role of individual factors in differential regulation motifs.

MCA has the advantage of being easy to compute and intuitively interpretable; it is in effect a moving window correlation analysis simultaneously over many window sizes. The MCA plot provides a graphical diagnostic for detection of subpopulations points that contribute inordinately to the overall correlation, or outliers, and may provide biological insights that serve as hypotheses for further experimentation. Finally, although we

have focused on biological data and in particular cellular subpopulations in single-cell transcriptional data, the method is more general and applicable to any multivariate data.

While the simplicity of MCA plots makes them easy to interpret, there are nonetheless shortcomings that must be mentioned. MCA plots are a graphical representation of the interaction of only two factors, sorted by a third. If there are many covariates, many such plots are possible, and it becomes increasingly more difficult to generate and search through all possible plots as the dimension increases. In such cases it is helpful to consider only those plots which may be of biological interest such as sorting variables thought to have a regulatory role, or pairs of factors that are suspected to interact. However, one may also use alternative sorting variables, such as products of covariates representing potential interactions, principal directions as determined by PCA, or even arbitrary non-linear functions of the covariates.

In the case of many variables, one may wish to sort the resultant plots according to arbitrary functions of the estimated correlation structures; i.e. one could filter for only those plots showing large significant regions or for plots for which a significant region of both positive and negative correlation are present. Although preliminary tests with such methods are successful in identifying such interesting plots, the results are not shown here as they are unnecessary when the number of dimensions is still manageable via manual inspection.

The correlation becomes difficult to estimate when the number of samples is small, or when the number of variables is relatively large compared to the number of observations. If the resolution is fine, then the MCA plot will contain many points for which the corresponding subpopulation only contains one or a few observations. Such points are omitted from the plot since the correlation cannot be robustly computed. This can sometimes give rise to small regions near the bottom of the MCA plots for which there are too few observations to compute the subpopulation correlation. These regions do not have biological significance.

Similarly, the stochastic nature of the data may give rise to "noise" in small subpopulations, leading to interspersed points on the MCA plot which are not part of a large, significant region. These points typically do not indicate robust subpopulations since a small perturbation away from them leads to a different correlation structure, and can safely be ignored. This "noise" also gives rise to the slight inhomogeneities in the regions identified in Figures 2 and 3.

Lastly, in the case of relatively many variables compared to the number of observations, correlations can be computed using shrinkage-based estimators [202], although this results in a different estimation of statistical significance, and increases computational complexity.

3.4 Conclusion

We have presented a method for the analysis of local correlation structures in subpopulations of multivariate data. MCA provides a multiresolution summary of correlations between pairs of variables as ordered by a third sorting variable. Using MCA, it is possible to detect robust correlations, identify outliers which can bias correlation estimates, and potentially discover new subpopulations or interactions giving rise to novel biological hypotheses.

Future work will focus on the development of methods to automatically identify variable pairs showing differential regulation in conjunction with a sorting variable, alleviating the need to manually search through plots for interesting behaviors.

3.5 Methods

We introduce Multiresolution Correlation Analysis (MCA) as a means for visually analyzing the local correlation structure of pairs of covariates, sorted by a sorting variable.

3.5.1 Estimation of correlations

The empirical estimation of the Pearson correlations of a pair of random variables is computed in the usual way, see 2.2.4.

If the data are not multivariate normally distributed, it is preferable to use a more robust measure of statistical correlation. For instance, Spearman's rank correlation coefficient is defined as in (2.9), but using the rank-transformed data [158]; it provides a non-parametric measure of correlation between a pair of covariates.

3.5.2 Multiresolution correlation analysis

We define the matrix

$$D = \begin{bmatrix} d_{11} & \dots & d_{1N} \\ \vdots & \vdots & \vdots \\ d_{1M} & \dots & d_{MN} \end{bmatrix} = \begin{bmatrix} \vec{d_1} & \vec{d_2} & \dots & \vec{d_N} \end{bmatrix}$$

as the matrix of observed data, where the rows correspond to individual observations, and columns to measured variables. Note that the data matrix is defined as the transpose of the data matrix employed in some other transcriptomic analysis methods.

Given D, we can compute the sample correlation between any pair of variables, for any subset of the total observations. In particular we examine subpopulations defined by different intervals within the distribution of $\vec{d_s}$, the s^{th} column of D, for any desired sorting variable s. For example, we can examine subpopulations for which the value of s is in the highest or lowest 30% of its distribution.

For a subpopulation centered on the α^{th} quantile of the sorting variable $\vec{d_s}$, and containing $\beta \times 100\%$, of the total observations, such that

$$\begin{aligned} 0 < &\quad \beta \quad \leq 1 \\ \frac{\beta}{2} < &\quad \alpha \quad < 1 - \frac{\beta}{2} \end{aligned} \tag{3.1}$$

we can compute the sample correlation matrix $\hat{\Sigma}(\alpha, \beta; s)$

$$\hat{\Sigma}(\alpha, \beta; s) = \{\hat{\sigma}_{ij}\}_{i,j=1\dots N} \tag{3.2}$$

with

$$\hat{\sigma}_{ij} = \widehat{cor}(\vec{d_i}(\alpha, \beta; s), \vec{d_j}(\alpha, \beta; s)) \tag{3.3}$$

and

$$\vec{d_q}(\alpha, \beta; s) = \left\{ d_{pq} \middle| Q(\alpha - \beta; s) \leq d_{pq} \leq Q(\alpha + \beta; s) \right\} \tag{3.4}$$

where Q is the quantile function, i.e. $Q(x; s)$ is the x^{th} quantile of the distribution of $\vec{d_s}$, and $\vec{d_q}(\alpha, \beta; s)$ is the subset of the q^{th} column of D for which the sorting variable falls between the $(\alpha - \beta)^{th}$ and $(\alpha + \beta)^{th}$ quantile of its distribution.

We define Ω to be the set of all pairs (α, β) for which Equation (3.1) is satisfied; for all $(\alpha, \beta) \notin \Omega$, $\hat{\Sigma}(\alpha, \beta; s)$ is undefined. Intuitively, Equation (3.1) constrains α and β such that the subpopulation can extend no lower than the minimum, and no higher than the maximum of the sorting variable.

Although any function could be computed for the subpopulations, we restrict ourselves to Pearson correlation. If there are relatively many variables compared to the number of observations, i.e. $N > M$, estimation of the correlation matrix becomes numerically infeasible. In this case, estimation of the correlation can be computed using shrinkage-based approaches such as implemented in the GeneNet R-package [202].

3.5.3 Construction of MCA plots

We systematically investigate the correlation of the subpopulations defined by $(\alpha, \beta) \in \Omega$. This information can be condensed into a MCA plot for any pair of variables (i, j) by plotting the magnitude of the $(i, j)^{th}$ entry of $\hat{\Sigma}(\alpha, \beta; s)$, with a color scale mapped to the interval $[-1, 1]$.

While $\hat{\Sigma}(\alpha, \beta; s)$ is in principle defined for all $(\alpha, \beta) \in \Omega$, in practice we choose $\beta = 1/R, \ldots, 0.5$ and $\alpha = \beta, \beta + 1/R, \ldots, 1 - \beta$ for some positive odd integer $R \leq M$ which determines the resolution of the MCA plot, i.e. the number of subpopulations examined: the larger R, the finer the resolution of the MCA plot.

For each computed subpopulation, a p-value is computed that depends on both subpopulation size and magnitude of the estimated correlation coefficient. Thresholding to retain only small p-values may reveal large subpopulations with strong correlations. However, due to the interdependence of the subpopulations (i.e. the estimated correlation coefficient of a subpopulation is determined by the correlation coefficients of the points below), it is not possibly to directly interpret the p-values as the probability of non-zero correlation.

Lastly, the number of possible MCA plots N increases cubically with the number of variables k, i.e. $N = k(k-1)(k-2)/6$, rendering fully-automatic analysis difficult. In this case, it is recommended to consider sorting variables which are of potential biological interest, such as those that are known to be heterogeneously expressed.

3.5.4 Implementation

MCA and the MCA plots were implemented using the `R` programming language. The routine allows the user to pass a data frame containing observations, select a sorting

variable, and a subset of factors whose pairwise correlations are to be analyzed; choose color options, and the number of subpopulations (resolution); specify correlation method (Pearson, partial, or Spearman), enable significance cutoffs with user-specified p-value threshold, and optionally to save resulting plots. The algorithm works by iterating through all subpopulations defined by median quantile of the sorting variable and size of the subpopulation, and computing the corresponding correlations using the built-in routines for correlation and significance estimation. Code is available upon request.

3.5.5 Stochastic simulation

Synthetic data were generated via simulation of a gene regulatory network, the dynamics of which obey a stochastic differential equation. Two cases were simulated: a three species activation model where of Z activates X, and X activates Y (Figure 3.1A, top); and an inhibition model for which Z activates X and X inhibits Y(Figure 3.1B, top).

The activation model obeys

$$\begin{aligned} \frac{dX}{dt} &= \frac{Z^{n_x}}{Z^{n_x} + {K_{zx}}^{n_x}} V_x - \beta_x \cdot X + \sigma \xi_X(t) \\ \frac{dY}{dt} &= \frac{X^{n_y}}{X^{n_y} + {K_{xy}}^{n_y}} V_y - \beta_y \cdot Y + \sigma \xi_Y(t) \\ \frac{dZ}{dt} &= k_z - \beta_z \cdot Z + \sigma \xi_Z(t) \end{aligned} \tag{3.5}$$

and the inhibition model obeys

$$\begin{aligned} \frac{dX}{dt} &= \frac{Z^{n_x}}{Z^{n_x} + {K_{zx}}^{n_x}} V_x - \beta_x \cdot X + \sigma \xi_X(t) \\ \frac{dY}{dt} &= \alpha_y + \frac{V_y}{X^{n_y} + {K_{xy}}^{n_y}} - \beta_y \cdot Y + \sigma \xi_Y(t) \\ \frac{dZ}{dt} &= k_z - \beta_z \cdot Z + \sigma \xi_Z(t) \end{aligned} \tag{3.6}$$

where model parameters are not necessarily the same between the activation and inhibition models.

In both cases, the drift of X is a sigmoidal function of Z and Z is an unregulated birth-death process. Each species is subject to linear decay and stochasticity enters through the homogeneous Wiener processes $\xi_X(t), \xi_Y(t), \text{and } \xi_Z(t)$ which are independent, with unit variance, and scaled by the factor σ.

The two systems were constructed in such a way that the steady state distributions do not fully overlap, but are instead displaced with respect to one another such that the inhibition model shows an approximately 40% increase in X, and 20% increase in Z with respect to the activation model.

Parameters and initial conditions used for the activation model are given in Table 3.1, and in Table 3.2 for the inhibition model. Simulations were performed using a Euler-Maruyama SDE integration scheme [169] with time step $\Delta t = 0.1$, implemented in MATLAB. The resulting simulations were allowed to converge to the steady state distribution by discarding the first 300 data points, and subsequently thinned by a factor of 20. Pearson correlations were computed using the `corr` built-in function of MATLAB.

3.5.6 Analysis of transcriptomic data

Single-cell transcriptomic data from 87 mouse embryonic stem cells were obtained from Trott, *et al.* [40] as an Excel spreadsheet containing qPCR readouts for eight pluripotency factors and one housekeeping gene. The expression of each gene was first adjusted by adding the minimum expression over all genes, 0.0217, and subsequently normalized by dividing by the expression of the gene *Gapdh* on a cell-wise basis.

Two cells were excluded due to the presence of missing data for some factors, and two additional cells were removed because they were thought to be outliers. The remaining 83 cells were subdivided into a Nanog+ compartment ($N = 20$), defined as the 20 cells with the highest Nanog expression, and for which no Fgf5 expression was detected, and the complementary Nanog- compartment ($N = 63$). The cells were separately divided into a Fgf5+ ($N = 15$) compartment, for which Fgf5 expression was detected, and a Fgf5- ($N = 68$) compartment with no Fgf5 expression.

Correlation networks were computed using Pearson correlation of the normalized data without any log transformation, and with a significance cutoff of 0.05.

3.6 Tables

Table 3.1: Model parameters used for activation model (Figure 3.1A).

Parameter	Value	Description
n_x	2	Hill coefficient of X activation
n_y	2	Hill coefficient of Y activation
K_{zx}	900	Equilibrium constant of X activation
K_{xy}	1000	Equilibrium constant of Y activation
V_x	600	Velocity of X production
V_y	600	Velocity of Y production
k_z	450	Basal production of Z
β_x	0.3	Death rate of X
β_y	0.3	Death rate of Y
β_z	0.5	Death rate of Z
X_0	100	Initial X
Y_0	100	Initial Y
Z_0	100	Initial Z
Δt	0.1	Time step

Table 3.2: Model parameters used for inhibition model (Figure 3.1B).

Parameter	Value	Description
n_x	2	Hill coefficient of X activation
n_y	2	Hill coefficient of Y activation
K_{zx}	4000	Equilibrium constant of X activation
K_{xy}	1000	Equilibrium constant of Y inhibition
V_x	10000	Velocity of X production
V_y	70	Velocity of Y production
k_z	110	Basal production of Z
a_y	70	Basal production of Y
β_x	0.5	Death rate of X
β_y	0.1	Death rate of Y
β_z	0.1	Death rate of Z
X_0	100	Initial X
Y_0	1500	Initial Y
Z_0	1000	Initial Z
Δt	0.1	Time step

Chapter 4

A case study on the use of scale separation-based analytical propagators for parameter inference in models of stochastic gene regulation

In order to provide an accurate, detailed model of Nanog regulation and expression dynamics, it is necessary to consider stochastic models which capture the time-dependent evolution of the Nanog promoter, mRNA and protein. However, even for such a small system it is not possible to compute the solution to the chemical master equation (CME) in closed form. Although it is in principle possible to compute the solution with little approximation error when the number of molecules is low (using the finite state projection (FSP), see Section 2.4.2), it rapidly becomes infeasible when molecule numbers become too large or when there are too many species involved. Thus, to investigate the time-dependent behavior of small regulatory motifs it is necessary to utilize approximations.

Shahrezaei *et al.* developed an approximation to the transition density of a regulatory motif involving a promoter that is always active, and mRNA and protein that are produced and degraded according to a birth-death process, where the probability of protein production depends on the mRNA copy number [111]. Recently, Popović *et al.* derived an extension to this model, based on geometric perturbation theory, which relaxes the assumption of infinite scale separation between mRNA and protein degradation, thus allowing for a better approximation of the time-dependent joint density even when the mRNA half-life is appreciable compared to the protein half-life [112]. In this chapter, I perform an investigation into the suitability of this approximation technique for parameter inference for this two-stage model. The goal of this investigation is to learn whether this method is appropriate for modeling Nanog dynamics and suitable for parameter identification. I further compare the newly developed method against the previous method of Shahrezaei, and against an approximation to the CME using the FSP. Although I did not develop the mathematical theory, I performed the entirety of the analysis described in this

chapter, including developing a numerically robust and efficient, C++-based implementation of the two approximate propagators. I also derived alternative formulations to some expressions (using identities of special functions) to render the computations feasible.

The work in this Chapter has been published in the following original research article:

> **Feigelman, J.**, Popović, N., Marr, C. (2015). A case study on the use of scale separation-based analytical propagators for parameter inference in models of stochastic gene regulation . Journal of Coupled Systems and Multiscale Dynamics, 3(2), 164173. http://doi.org/10.1166/jcsmd.2015.1074

The text and figures are entirely my work, with minor corrections from co-authors.

4.1 Introduction

Gene expression is a complex and highly regulated multi-step process responsible for the timely synthesis of proteins necessary for cellular function. At the molecular level, gene expression is inherently stochastic due to random binding events of transcription factors and the transcriptional machinery, which ultimately leads to mRNA transcription with probabilities depending on the concentration of the reaction educts. Protein synthesis requires a chance encounter of mRNA with ribosomes, and mRNA or protein degradation an encounter with the degradation machinery. Thus, models for gene expression have to capture the stochasticity at both mRNA and protein levels.

A simple, "two-stage" model for stochastic gene expression consists of a constitutively active gene from which an mRNA molecule can be transcribed, and protein, the production of which depends on the instantaneous abundance of mRNA (see Figure 4.1A). Both mRNA and protein are subjected to stochastic degradation. Such a qualitative model can be described mathematically as a two-dimensional Markov jump process in the copy numbers of mRNA and protein, with reaction probabilities that are functions of the current state only (hence the Markov property), and suitably chosen kinetic constants [105, 111].

While the two-stage model is easily simulated using stochastic simulation algorithms such as Gillespie's algorithm [117], it is nonetheless a difficult task to derive analytical expressions for the evolution of mRNA and protein copy numbers with time. The Markov process itself obeys the chemical master equation (CME), an infinite-dimensional system of ordinary differential equations, for which no exact (closed-form) solutions are known in general. Numerous approaches exist for approximately solving the CME such as the linear noise approximation [167], a second-order Taylor series expansion in the system size of the reaction volume; moment equations and variants [98, 203], which capture an arbitrary number of statistical moments of the stochastic process; finite state projection [131], a truncation of the state space of possible copy number combinations, and many others (for an overview, see [92]). We further note that this particular system has been studied using a variety of analytical and computational techniques, see e.g. [42, 105, 113, 204] or [109] for a review of related modelling approaches for this system.

An alternative analytical approach was developed by Shahrezaei and Swain [111], wherein it is assumed that mRNA molecules decay much faster than protein, a realistic assumption in many prokaryotic cells. In the limit of a perfect scale separation in which the decay of mRNA is instantaneous, the CME underlying the two-stage model can

be solved analytically by the introduction of a generating function. The latter then obeys a first order linear partial differential equation, the solution of which can be obtained via the method of characteristics. The resulting analytical expression for the general time-dependent joint probability density of mRNA and protein, called the propagator of the system, is of great utility for understanding the time-dependent behavior of the system. However, it is not valid when the assumption of scale separation is violated as is commonly the case for eukaryotic cells. In recent work [112], the procedure developed in [111] was extended to capture departure from the assumption of perfect scale separation: the ratio of degradation rates of protein and mRNA, denoted ε, was taken to be small and positive instead of zero, as was the case in [111]. The presence of the (singular) perturbation parameter ε allows for the application of asymptotic techniques, such as geometric singular perturbation theory [205] and matched asymptotic expansions [206].

In the present case study, we explore the utility of this newly developed perturbative approach for propagator-based parameter inference in systems with varying degrees of scale separation. Specifically, the goal is to estimate molecular parameters in the model from observations of protein abundance only. Trajectories are simulated via Gillespie's stochastic simulation algorithm in a parameter regime in which mRNA and protein are produced continuously, i.e. not in translational bursts. The protein time-courses are sampled at regular time intervals, thus mimicking a typical time-lapse fluorescence microscopy setup [207, 208]. While fluorescence microscopy yields only intensity time courses, these can nonetheless be converted into absolute protein numbers if a calibration factor of molecules per unit intensity can be estimated, see e.g. [209]. mRNA time-courses are not observed, and hence are not used for parameter inference.

The zeroth-order propagator obtained by setting $\varepsilon = 0$ [111] is then compared to a first order propagator (in $\varepsilon > 0$) that is uniformly valid both on short and on long time-scales [112], in terms of the ability of each to capture the correct parameters – i.e., the kinetic constants in the underlying chemical reaction network – in the two-stage model for gene expression. A number of simplifying assumptions are made; notably, we ignore impeding factors such as measurement noise, uncertainty in the conversion from fluorescence intensity to protein numbers or low sampling frequency of fluorescent signal. Rather, our focus in this case study is on assessing the general efficiency and accuracy of the propagator-based approach for parameter estimation.

4.2 Methods

4.2.1 Two-stage Gene Expression Model

We model gene expression as a two-stage process, whereby DNA is transcribed to mRNA, which is then translated into protein (see Figure 4.1A). We denote the probability of n molecules of protein and m of mRNA in the system at time τ by $P_{m,n}(\tau)$, i.e. $P_{m,n}(\tau) = P(M = m, N = n, t = \tau)$, where M, and N denote the number of mRNAs and proteins, respectively. The probability density $P_{m,n}(\tau)$ evolves according to the non-dimensionalized

CME (see B.2) [111, 167]

$$\begin{aligned}\frac{\partial P_{m,n}}{\partial \tau} &= a(P_{m-1,n} - P_{m,n}) + \gamma b m(P_{m,n-1} - P_{m,n}) \\ &\quad + \gamma[(m+1)P_{m+1,n} - mP_{m,n}] \\ &\quad + [(n+1)P_{m,n+1} - nP_{m,n}]. \end{aligned} \tag{4.1}$$

Here, m and n denote mRNA and protein copy numbers, respectively, a is the non-dimensional transcription rate and b is the non-dimensional translation rate, while the degradation rates of mRNA and protein are given by γ and 1, respectively (see Figure 4.1A). Finally, τ denotes a suitably non-dimensionalized time variable.

As in [111, 112], we define the perturbation parameter $\varepsilon = \gamma^{-1}$ here. It follows that for ε sufficiently small, the dynamics of Eq. (4.1) will vary on two distinct time-scales: the long-term behavior of the system is naturally described on the "slow" τ-scale, while the "fast" transients evolve according to the rescaled time $t := \frac{\tau}{\varepsilon}$.

4.2.2 Propagator Expressions

In this section, we collect a number of analytical results that underlie the present case study; details can be found in [111, 112].

Zeroth-Order Propagator

The zeroth-order propagator for the two-stage gene expression model (Figure 4.1A) represents an approximation to the CME, Eq. (4.1), under the assumption of infinitely fast mRNA degradation. Mathematically speaking, it is obtained in the singular limit of $\gamma \to \infty$, i.e., of $\varepsilon \to 0$. Following [111], we have

$$\begin{aligned}P_{n|n_0}(\tau, 0) = (1 - \mathrm{e}^{-\tau})^{n_0} \Big(\frac{1 + b\mathrm{e}^{-\tau}}{1+b}\Big)^a \Big(\frac{b}{1+b}\Big)^n \sum_{k=0}^{n} \Bigg\{ \frac{(-1)^k}{k!(n-k)!} \frac{\Gamma(a+n-k)\Gamma(n_0+1)}{\Gamma(a)\Gamma(n_0-k+1)} \\ \times \Big[\frac{1+b}{b(1-\mathrm{e}^{\tau})}\Big]^k {}_2F_1\Big(-n+k, -a, 1-a-n+k, \frac{1+b}{\mathrm{e}^{\tau}+b}\Big)\Bigg\}\end{aligned} \tag{4.2}$$

for the zeroth-order marginal probability $P_{n|n_0}(\tau, 0)$ of observing n protein molecules after time τ, given n_0 molecules of protein and $m_0 = 0$ of mRNA initially. Here, ${}_2F_1(a, b, c, z)$ is the Gauss hypergeometric function [210], see Appendix A.6. We remark that, by construction, $P_{n|n_0}(\tau, 0)$ neglects any contributions from the fast t-scale, as the decay of mRNA is instantaneous to leading order in ε.

Uniform (First-Order) Propagator

The uniform propagator, denoted $\mathcal{P}_{n|n_0}(\tau, t, \varepsilon)$, was derived as in [112]. Here, ε denotes the perturbation parameter, as before, while t is the fast time variable. We emphasize that $\mathcal{P}_{n|n_0}$ describes the probability of transitioning from n_0 protein molecules initially to

n at time $\tau = \varepsilon t$, uniformly on the two time-scales. After some algebraic rearrangement, we find

$$\begin{aligned}\mathcal{P}_{n|n_0}(\tau,t,\varepsilon) &= P_{n|n_0}(\tau,\varepsilon) + \varepsilon a\Big(\frac{b}{1+b}\Big)^{n-n_0}\frac{1}{(1+b)^2}\\ &\times [n-n_0-b-(1+b)t] + \frac{\varepsilon a}{\Gamma(n-n_0+2)}(bt)^{n-n_0}t\\ &\times \Big\{ {}_1F_1(n-n_0+1, n-n_0+2, -(1+b)t)t\\ &\times \Big[1-\frac{(n-n_0-b)}{1+b}\Big] + \frac{(n-n_0+1)}{1+b}\mathrm{e}^{-(1+b)t}\Big\} \end{aligned} \tag{4.3}$$

to first order in ε; here, ${}_1F_1(a,b,z)$ is the Kummer function of the first kind (or confluent hypergeometric function) [210]. We remark that the transition probability $P_{n|n_0}(\tau,\varepsilon)$ contributes on the slow τ-scale in Eq. (4.3), while the t-dependent contribution in Eq. (4.3) accounts for the transient dynamics on the fast time-scale.

Specifically, $P_{n|n_0}(\tau,\varepsilon)$ denotes the marginal probability, up to and including $\mathcal{O}(\varepsilon)$-terms, of observing n protein molecules after time τ given n_0 protein and $m_0 = 0$ mRNA molecules initially:

$$P_{n|n_0}(\tau,\varepsilon) = \sum_{m=0}^{\infty} P_{m,n|0,n_0}(\tau,\varepsilon) \tag{4.4}$$

As shown in [112], the probability of having more than 1 molecule of mRNA at time τ is negligible; thus, Eq. (4.4) reduces to

$$P_{n|n_0}(\tau,\varepsilon) = P_{0,n|0,n_0}(\tau,\varepsilon) + P_{1,n|0,n_0}(\tau,\varepsilon). \tag{4.5}$$

After some algebraic simplification, the two transition probabilities $P_{0,n|0,n_0}$ and $P_{1,n|0,n_0}$ in the above relation are found to be

$$\begin{aligned} P_{0,n|0,n_0}(\tau,\varepsilon) &= (1-\mathrm{e}^{-\tau})^{n_0}\Big(\frac{b}{1+b}\Big)^n\Big(\frac{1+b\mathrm{e}^{-\tau}}{1+b}\Big)^a\\ &\times\sum_{k=0}^{n}\frac{1}{(n-k)B(a,n-k)}{}_2F_1\big(-n+k,-a,1-a-n+k,\tfrac{1+b}{\mathrm{e}^{\tau}+b}\big)\\ &\times\Big\{g(n_0,k) - \frac{\varepsilon}{2}\frac{a}{(1+b)^2}(k+1)\times\Big[{}_2F_1\big(-k,-n_0,-1-k,\tfrac{1+b}{b(1-\mathrm{e}^{\tau})}\big)\\ &+\Big(\frac{1+b}{\mathrm{e}^{\tau}+b}\Big)^{k+2}\mathrm{e}^{2\tau}{}_2F_1\big(-k,-n_0,-1-k,\tfrac{\mathrm{e}^{\tau}+b}{b(1-\mathrm{e}^{\tau})}\big)\Big]\Big\}\\ g(n_0,k) &= \begin{cases} 0 & \text{for } k>n_0\\ (-1)^k\binom{n_0}{k}\big[\frac{(1+b)}{b(1-\mathrm{e}^{\tau})}\big]^k & \text{for } k\le n_0\end{cases}\end{aligned} \tag{4.6}$$

$$
\begin{aligned}
P_{1,n|0,n_0}(\tau,\varepsilon) &= a\varepsilon\Big(\frac{b}{b+1}\Big)^n \frac{1}{b+1}(1-\mathrm{e}^{-\tau})^{n_0}\Big(\frac{1+b\mathrm{e}^{-\tau}}{1+b}\Big)^a \\
&\quad\times \sum_{k=0}^{n}\Bigg\{\frac{1}{(n-k)B(a,n-k)}{}_2F_1\big(k-n,-a,-a+k-n+1,\tfrac{b+1}{b+\mathrm{e}^{\tau}}\big) \\
&\quad\times\Bigg[h(n_0,k)+(-1)^{n_0}\Big[\frac{b\mathrm{e}^{\tau}+1}{b(1-\mathrm{e}^{\tau})}\Big]^{n_0}\Bigg]\Bigg\} \\
h(n_0,k) &= \begin{cases} 0 & \text{for } k\geq n_0 \\ \binom{n_0}{k+1}\big[\frac{b+1}{b(1-\mathrm{e}^{\tau})}\big]^{k+1}{}_2F_1\big(1,k-n_0+1,k+2,\frac{b+1}{b(1-\mathrm{e}^{\tau})}\big) & \text{for } k<n_0 \end{cases}
\end{aligned}
\tag{4.7}
$$

Here, $B(a,b) := \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ is the Beta function, with the proviso that $\frac{1}{(n-k)B(a,n-k)} = 1$ when $n = k$.

Finally, Eq. (4.3) can be simplified by substituting

$$
\begin{aligned}
&{}_1F_1(n-n_0+1;n-n_0+2;-(1+b)t) = \\
&\quad[(1+b)t]^{-(n-n_0+1)}\Gamma(n-n_0+2) \\
&\quad-(n-n_0+1)\Gamma\big(n-n_0+1,(1+b)t\big)
\end{aligned}
\tag{4.8}
$$

to achieve the computationally more tractable formulation

$$
\begin{aligned}
&\mathcal{P}_{n|n_0}(\tau,t,\varepsilon) = P_{n|n_0}(\tau,\varepsilon) \\
&+\varepsilon a\Big(\frac{b}{1+b}\Big)^{n-n_0}\frac{1}{(1+b)^2}[n-n_0-b-(1+b)t] \\
&+\varepsilon at\Bigg\{-\Big(\frac{b}{1+b}\Big)^{n-n_0}\frac{1}{(1+b)t}\Big(\frac{b+n_0-n}{1+b}-t\Big) \\
&\times\big[1-Q(n-n_0+1,(1+b)t)\big] \\
&\quad+\frac{(bt)^{(n-n_0)}}{1+b}\frac{\mathrm{e}^{-(1+b)t}}{\Gamma(n-n_0+1)}\Bigg\}.
\end{aligned}
\tag{4.9}
$$

Here, $Q(a,x) := \frac{\Gamma(a,x)}{\Gamma(a)}$ denotes the regularized upper incomplete gamma function.

4.2.3 Special Cases of the Hypergeometric Functions

Care must be taken when evaluating the hypergeometric function ${}_2F_1(a,b,c,z)$. The following special cases are of use [210].

- $a = -k = c$ ($k \in \mathbb{Z}^+$):

$$
\begin{aligned}
{}_2F_1(a,b,c,z) &= {}_2F_1(-k,b,-k,z) \\
&= \sum_{n=0}^{m}(b)_n\frac{z^n}{n!},
\end{aligned}
\tag{4.10}
$$

where $(x)_n = x(x+1)\dots(x+n-1)$ is the rising factorial of x.

- $a = -k$, $c = -k - 1$ ($k \in \mathbb{Z}^+$):

$$\begin{aligned} {}_2F_1(a, b, c, z) &= {}_2F_1(-k, b, -k - 1, z) \\ &= \sum_{n=0}^{\min(-a,-b)} (b)_n \frac{z^n}{n!} \frac{a + n - 1}{a - 1}. \end{aligned} \tag{4.11}$$

- $a > 0$, $c > 0$, $b = -k$ ($k \in \mathbb{Z}^+$):

$$\begin{aligned} {}_2F_1(a, b, c, z) &= {}_2F_1(a, -k, c, z) \\ &= \sum_{n=0}^{k} \frac{(a)_n (b)_n}{(c)_n} \frac{z^n}{n!}. \end{aligned} \tag{4.12}$$

4.2.4 Stochastic Simulation

Stochastic simulations were performed using the StochKit 2.0 [211] simulation framework and the standard stochastic simulation algorithm [117], with a non-dimensionalized transcription rate $a = 20$ and a non-dimensionalized translation rate $b = 2.5$, corresponding to 'regime I', i.e. bursty mRNA synthesis, as defined in [112]. We used mRNA degradation rates of $\gamma \in \{10, 20, 50, 100\}$ and protein degradation rate 1. Each value of γ was simulated 20 times, and the resulting trajectories were used for computing the probability landscapes of the rescaled model parameters a and b. Protein quantities were observed without measurement noise at intervals of 0.1 time units. All simulation runs assumed zero molecules of mRNA and protein initially, i.e., $m_0 = 0 = n_0$.

4.2.5 Implementation

Both the zeroth-order propagator, Eq. (4.2), and the uniform propagator Eq. (4.3), were implemented in C++ with a Matlab mex-file interface. Special functions were evaluated using the GNU scientific library [212], the `Hyp_2F1` function implementation of the Gauss hypergeometric function [213], and the `Algorithm 910` multiprecision special function library [214]. It proved indispensable to use a high precision numerical library due to several computations involving subtraction of very large numbers. While the difference of such numbers is potentially below a double precision machine error of approximately 10^{-13}, they are nonetheless essential in the correct computation of the transition probabilities. However, our C++ implementation is still inaccurate in some extreme cases, typically for very large protein numbers n, due to numerical differences which are sometimes as small as 10^{-370} in Eq. (4.7), but which unfortunately cannot be neglected as they are inflated by the remaining terms in the expression. Such inaccuracies are infrequent, though, and generally occur during transitions for which the uniform propagator yields non-physical values; thus, they do not substantially affect our analysis, or the conclusions obtained in this study.

4.3 Results and Discussion

To assess the applicability of the zeroth-order propagator Eq. (4.2), $P_{n|n_0}(\tau, 0)$, and the uniform propagator Eq. (4.3), $\mathcal{P}_{n|n_0}(\tau, t, \varepsilon)$, for parameter inference in the two-stage gene

expression model, we simulate time series with a specific parameter pair (a^*, b^*). Then, we compute the likelihood of the observed dataset on the basis of the two propagators for a range of values for the parameters a and b. For simplicity, we assume the scale separation γ between mRNA and protein lifetimes to be known (see Methods for definitions).

4.3.1 Protein Time Courses Simulated With Gillespie's Algorithm

We simulate mRNA and protein time-courses for the two-stage gene expression model (Figure 4.1A) using Gillespie's algorithm [117] (see Methods for details). Simulations are initialized with $m_0 = 0$ mRNA molecules, and $n_0 = 0$ protein molecules, although the method is equally applicable to any initial number of proteins, as shown in [112].

The generated protein time-courses are sampled at $N = 101$ timepoints, with fixed time increments of $\Delta t = 0.1$ to mimic the measurement of protein abundance with time-lapse microscopy, see Figure 4.1B. For each transition in the observed time series, we compute the approximate probability using the two analytical propagators, see Figure 4.1B, inset. Notably, we ignore measurement noise throughout, i.e., we only investigate the suitability of the derived propagator expressions for synthetic "ideal" data (see Discussion for possible extensions).

We note, moreover, that the propagator expressions can be used to visualize the likelihood of various sample paths in the underlying stochastic networks for a given set of parameters and conditional on the initial condition; see Figure 4.1C.

4.3.2 Parameter Inference

Using the two analytical propagators, we compute the log-likelihood $L(a, b)$ of the simulated trajectories for a range of (a, b) combinations in the subspace $(a, b) \in [10^{-1}, 10^3] \times [10^{-3}, 10^3]$. The log-likelihood is computed as

$$L(a, b) = \sum_{i=1}^{N} \log P^*_{n_i|n_{i-1}}, \tag{4.13}$$

where $P^*_{n_i|n_{i-1}} = P_{n_i|n_{i-1}}$ given by (4.2) for the zeroth-order propagator and $P^*_{n_i|n_{i-1}} = \mathcal{P}_{n_i|n_{i-1}}$, given by (4.9) for the uniform propagator, where the parameter $\epsilon = \gamma^{-1}$ is assumed to be known; both propagators depend on the parameters a and b. The term n_i represents the number of proteins at measurement time t_i. Thus, we compute the logarithm of the probability of each transition, from n_{i-1} protein molecules at time t_{i-1} to n_i molecules at time t_i, in the sequence of observed measurements (see Figure 4.1B, inset).

In order to estimate the parameters a and b from simulated protein time-courses, Eq. (4.13) has to be evaluated very frequently. We thus developed a numerically stable expression for the uniform propagator $\mathcal{P}_{n|n_0}$ (see Section 4.2.2), and we used an efficient implementation in C++ for both propagators that results in reasonable runtimes; see Section 4.2.5 for details.

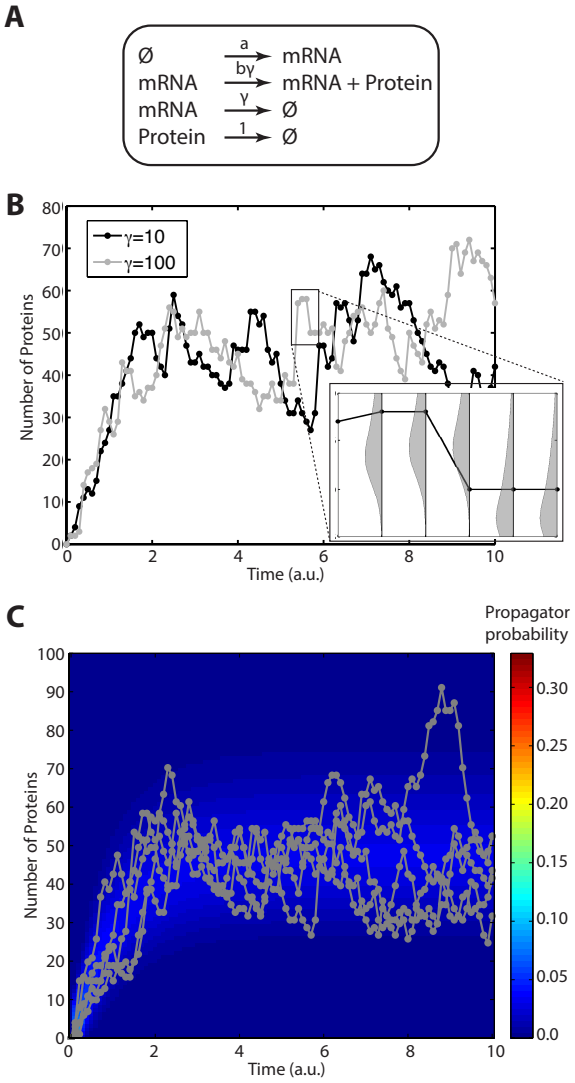

Figure 4.1: A. The two-stage model for gene expression, which captures stochastic mRNA and protein birth and death, with non-dimensionalized parameters a for transcription, b for translation, and γ for mRNA degradation. B. Time-courses were simulated using the stochastic simulation algorithm, shown here for $a = 20, b = 2.5, \gamma = 10$ and $\gamma = 100$. Probabilities can be computed for each protein transition using the analytical expression for the two-stage propagators, Eq. (4.9) and Eq. (4.2) (inset, probability distributions shown in gray). C. Propagator expressions can be used to compute the probability a particular number of protein molecules at arbitrary timepoints, conditional on the initial conditions. The prediction from the uniform propagator, Eq. (4.3) (blue background), shows good qualitative agreement with stochastic simulation (gray lines), shown for $\gamma = 10$.

4.3.3 Comparison of Propagator Accuracy and Efficiency

We scan the space of parameter values (a, b) on a logarithmically spaced 44×45-grid with $10^{-1} \leq a \leq 10^3$ and $10^{-3} \leq b \leq 10^3$. For each pair (a, b), we compute the log-likelihood $L(a, b)$ of the zeroth-order propagator, resulting in a likelihood landscape that should ideally have its maximum, the maximum likelihood estimator (MLE), at the true parameter values. We immediately encountered the obstacle that the uniform propagator yields negative transition probabilities, or even probabilities larger than one, for some choices of (a, b). This is discussed in [112], and is a result of the asymptotic approximation. Nonetheless it is problematic when computing the overall log-likelihood as the computation (4.13) becomes meaningless. Thus we introduce an 'averaged log-likelihood', $\bar{L}(a, b)$, which removes all non-physical values (i.e. larger than one or less than or equal to zero):

$$\bar{L}(a,b) = \frac{\sum_{i=1}^{N} \mathbb{1}\left\{0 < P^*_{n_i|n_{i-1}} \leq 1\right\} \left[\log P^*_{n_i|n_{i-1}}\right]}{\sum_{i=1}^{N} \mathbb{1}\left\{0 < P^*_{n_i|n_{i-1}} \leq 1\right\}} \tag{4.14}$$

with $P^*_{n_i|n_{i-1}}$ defined as in (4.13). The averaged log-likelihood represents the average log-likelihood for a set of parameters (a, b), after removing all non-physical transition densities. The averaging is used to compensate for the fact that the number of non-physical transitions may vary greatly for different (a, b). Since each retained transition only decreases the overall log-likelihood of the time series, the log-likelihood estimate without normalization would inherently be biased towards regions of (a, b)-space for which many transitions were omitted.

Using (4.14), we compute the log-likelihood landscapes (shown as contour plots) for the zeroth-order and uniform propagators, obtained from a single time-course simulated with $\gamma = 100$, observed at $N = 101$ timepoints at time intervals of $\Delta t = 0.1$. Computing the MLE, we find that it deviates from the true model parameters $(a^*, b^*) = (20, 2.5)$ in (a, b)-space, for both the zeroth-order propagator (Figure 4.2A), and the uniform propagator (Figure 4.2B). For comparison, we also generate the finite state projection approximation to the log-likelihood landscape (Figure 4.2C), computed by solving the CME (4.1), assuming that the mRNA has at most two copies (in agreement with simulations), and the the number of proteins does not exceed 200, see [131] for details of the FSP. Similar to the two approximate propagators, the FSP shows a bias in the MLE, from which we can conclude that the bias originates largely due to the inherent stochasticity of the system.

For all three methods, the MLE converges to the true model parameters (a^*, b^*) as the number of simulations used for parameter inference increases from one to twenty, see Figure 4.2D-F, wherein we depict the sum of the averaged log-likelihoods over each of the trajectories. Thus, we conclude that for $\gamma = 100$, both analytical propagators provide a good approximation to the underlying transition density, and may be of use for parameter inference. However, the finite state projection provides a log-likelihood landscape that is more tightly peaked around the true model parameters (compare contour lines, in Figure 4.2D-F); the approximate methods are less able to to distinguish between combinations of a and b which lead to approximately the same dynamics in the observed time courses.

Although both approximate propagators work well when γ is sufficiently large, and thus for which the perturbation parameter $\epsilon = \gamma^{-1}$ is small, problems emerge when moving to

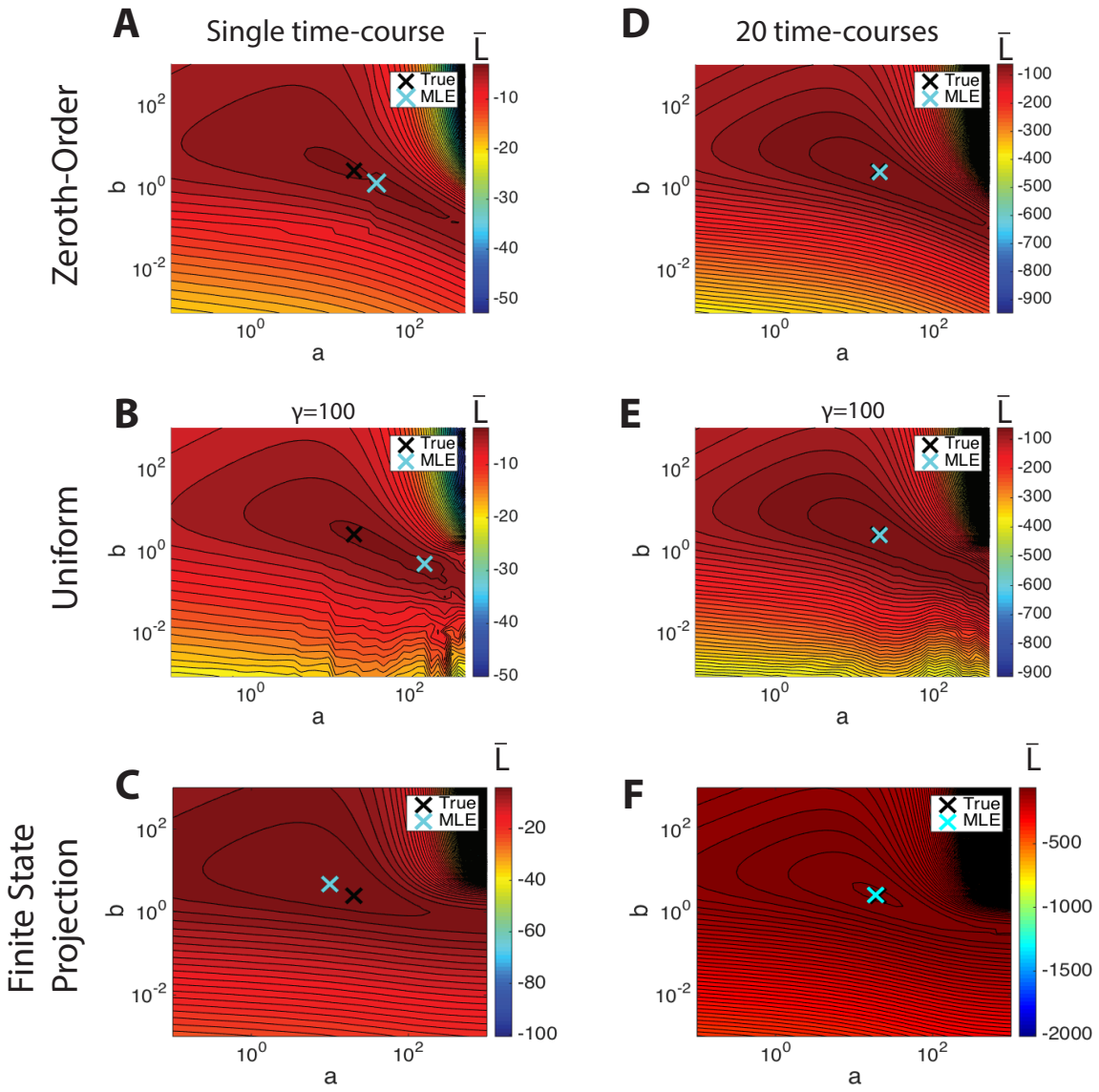

Figure 4.2: Averaged log-likelihood landscapes for simulations with $(a, b) = (20, 2.5)$ and $\gamma = 100$. Landscapes for single-time courses (left) are shown with contour lines drawn at intervals of 1 unit; contours for landscapes using 20 time-courses (right) are drawn at intervals of 10 units. The averaged log-likelihood landscapes generated using a single time-course for the zeroth-order propagator, Eq. (4.2), (A), uniform propagator, Eq. (4.3), (B), and finite state projection approximation (C) for a single time-course each show bias of the MLE with respect to the true model parameters. Notably, the landscape of the uniform propagator (B) shows distortions arising from non-physical transitions probabilities for some parameters (a, b). As the number of trajectories is increased to 20, the MLE converges to the true parameters for each of the zeroth-order propagator (D), uniform propagator (E), and the finite state projection (F). The averaged log-likelihood of the finite state projection seems to be the most tightly-peaked around the true model parameters.

smaller γ. In the case of $\gamma = 10$, the uniform propagator generates many non-physical transition probabilities which heavily distorts the log-likelihood landscape, see Figure 4.3A. This leads to a severe bias of the MLE with respect to the true model parameters.

To understand the origins of this bias, it is helpful to examine a representative time-course. In Figure 4.3B, a typical protein time-course with $\gamma = 10$ is shown (top), along with the log-likelihood of the transitions (bottom) obtained using the uniform propagator for the true parameter values (black) and for the MLE (cyan). The transitions for which the uniform propagator, Eq. (4.3), yields non-physical values are shown as white squares within the colored bars at the bottom of Figure 4.3B. We indicate one such transition using arrows in Figure 4.3B, and compute the corresponding transition probability distribution using the uniform propagator, Figure 4.3C. In this example, the protein time-course transitions from 55 to 57 molecules within one time interval. Examining the propagator evaluated for the true parameters (a^*, b^*) with initially 55 protein molecules, i.e., calculating $P_{57|55}$, we see that the propagator becomes negative for $57 \leq n \leq 60$ (Figure 4.3C, arrow). We note that the corresponding negative values are of order $\mathcal{O}(\gamma^{-2})$, and thus within the error incurred by the expansion in Eq. (4.3), which is accurate to $\mathcal{O}(\gamma^{-1})$.

Using the uniform propagator, we compute a portion of the "transition matrix", i.e. the probability of all transitions from $0 \leq n_{i-1} \leq 100$ to $0 \leq n_i \leq 100$, evaluated at the true model parameters $(a^* b^*)$, see Figure 4.3D. Using this plot it is obvious that large regions of the transition space yield non-physical values, shown in gray.

To quantify the frequency of the non-physical transitions, we calculate a "computability score"

$$C(a,b) = \frac{1}{N \cdot N_{\text{traj}}} \sum_{k=1}^{N_{\text{traj}}} \sum_{i=1}^{N} \mathbb{1}\left\{0 < P_{n_i^k | n_{i-1}^k} \leq 1\right\}, \tag{4.15}$$

where the superscript k indicates the index of the simulated trajectory. Thus $C(a,b)$ captures the fraction of evaluated transitions for a given pair (a,b) which were physically admissible, i.e., between zero and one, for the uniform propagator. A plot of the computability score reveals that certain regions of the parameter space suffer from low computability, i.e., they yield many non-physical values, which are apparent as dark regions, see Figure 4.3E. By contrast, the uniform propagator provides a better approximation to the true transition probability when evaluated in the so-called regime II $(a,b) = (0.5, 100)$ (as defined in [112]), which corresponds to a continual protein synthesis. This is obvious when examining the transition matrix, Figure 4.3F, for which all transitions were computable and physically admissible, as opposed to the transition matrix for $(a,b) = (20, 2.5)$.

Thus we conclude that the uniform propagator may provide a useful approximation to the stochastic propagator in certain regions of parameter space, in particular for high b, low a corresponding to regime II, but breaks down in large regions of parameter space for which the computability is low. In regions with low computability, the remaining transitions may in fact have a higher likelihood than for the true model parameters (see Figure 4.3B) which can lead to a biased estimate of the model parameters, as in Figure 4.3A.

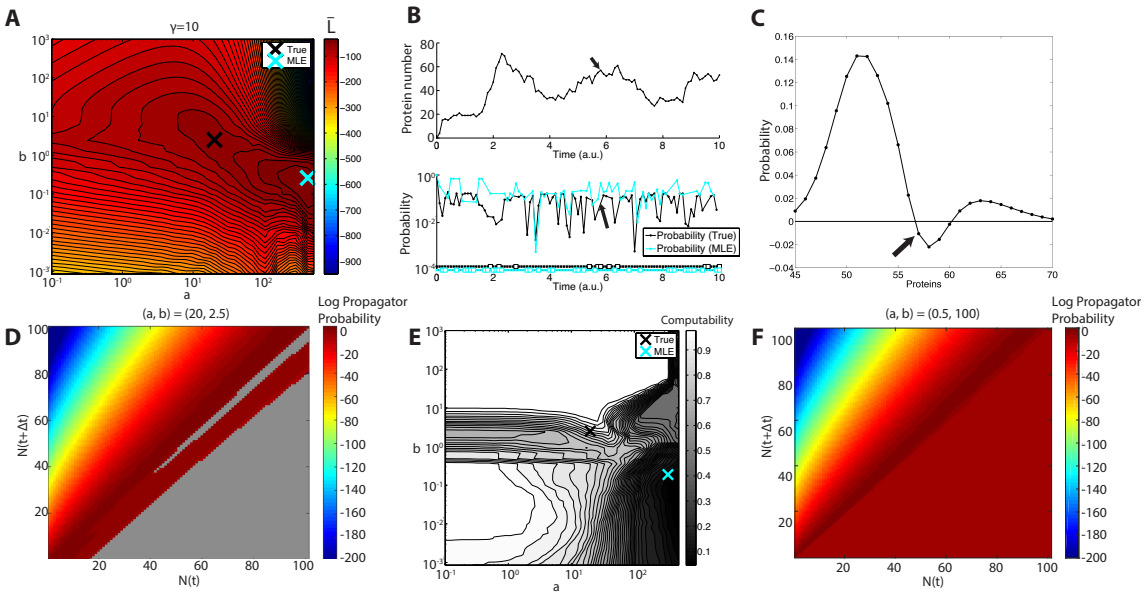

Figure 4.3: A. The averaged log-likelihood landscape for $\gamma = 10$ of the uniform propagator shows prominent distortions in the contours caused by frequent non-computable transitions. The MLE (cyan) shows an obvious bias with respect to the true model parameters (black). B. Inspection of a single time-course (shown on top) evaluated at the true model parameters and at the MLE reveals more non-computable transitions (indicated with white boxes below) for the MLE than for the true parameters; however, for those points that can be computed, the probability is higher than for the true parameters, leading to a higher averaged probability and thus a biased estimate of the parameters (a, b). C. Inspection of the transition probability with $(a, b) = (20, 2.5)$ for the transition marked with arrows in (B), from 55 to 57 molecules, reveals a negative transition probability coinciding with the observed transition. D. Computation of the transition matrix for the uniform propagator ($(a, b) = (20, 2.5)$) from $N(t)$ proteins to $N(t + \Delta t)$ proteins reveals a large region of non-computable transitions, shown in gray. E. The computability score $C(a, b)$ shows that the MLE is biased towards the region with the lowest computability, for which most transitions are omitted from the averaged log-likelihood score $\bar{L}(a, b)$. F. By contrast, the transition matrix is fully computable for $(a, b) = (0.5, 100)$ corresponding to the region of continuous protein synthesis, i.e. non-bursty dynamics.

4.4 Conclusion

In this work, we have investigated the utility of a propagator-based approach for approximating the transition probabilities in a simple two-stage gene expression model by attempting parameter inference from protein time-courses. The latter can be derived, e.g., from time-lapse microscopy of fluorescently-labelled proteins in single cells, and are thus of interest for the study of regulation in gene expression. Here, we only use simulated time-courses measured at regular intervals, without measurement noise. The simulations are initialized with zero mRNA molecules and zero protein molecules. This represents a simplifying assumption as compared to a typical biological setting, but does not affect the subsequent analysis.

We compare a newly developed uniform propagator, which was derived in [112] by application of geometric singular perturbation techniques, to a previously proposed propagator [111] which corresponds to the singular limit as the perturbation parameter in the model is decreased to zero. The two propagators are compared on the basis of the probability landscapes of the two relevant model parameters a and b, which represent rescaled transcription and translation, respectively. The propagators are further compared against another approximate solution of the chemical master equation (4.1), corresponding to the finite state projection (FSP). The FSP is a numerical method and is *a priori* restricted to a subspace of the possible configurations of the system; nonetheless, it shows very good identifiability of the model parameters given sufficiently many observed trajectories (see Figure 4.2F).

The results of our investigation indicate that both propagators perform well when the value of γ — the non-dimensionalized mRNA degradation rate — is sufficiently large. In the case of $\gamma = 100$, both capture the true model parameters almost exactly, as long as there are sufficiently many time-courses. In our simulations, 20 time-courses — about 2000 observed transitions — were needed before convergence to the true parameter values, a number which is attainable in a real biological experiment. However, for smaller values of γ, that is, assuming a decrease in scale separation between mRNA and protein degradation, the uniform propagator becomes inconsistent, in that it generates negative transition probabilities for many segments of the protein time-course. This loss of positivity is a general feature of asymptotic expansions for probability distributions, which *a priori* only satisfy the non-negativity required of the distributions provided the corresponding perturbation parameter is sufficiently small. While the occurrence of negative probabilities for transient times, i.e., on the fast time-scale, is irrelevant for the evaluation of the steady state of the system, it is of extreme relevance to the utility of the propagator for parameter inference. Thus although the zeroth-order propagator is asymptotically less accurate due to the exclusion of the correction term, it nonetheless may prove a more useful approximation when used for parameter inference, as it does not yield negative transition densities under any circumstances.

Since the majority of time-courses contained transitions for which the calculated probabilities were negative, it was necessary to devise a better measure which utilized as much information as possible. We thus discard all negative transitions, and use the remaining non-negative transitions normalized by the number of non-negative transitions in each time-course to obtain an averaged likelihood for each pair (a, b) in the parameter space.

While this approach retains the maximum information possible from the trajectories, it nonetheless seemingly introduces distortions into the probability landscapes of the parameter space (see Figure 4.3A). These distortions proved sufficient to shift the MLE away from the true value, thus limiting the utility of the uniform propagator for parameter inference in this regime.

In the current analysis, we have restricted ourselves to computing the log-likelihood landscape, i.e., the approximate averaged log-likelihood $\bar{L}(a, b)$, for all parameter pairs (a, b) on a discrete grid that was sampled uniformly in log-space (see Methods). This approach is useful for visualizing the probability landscape, but is not ideal for parameter inference. In a more realistic setting, one would compute the maximum likelihood estimator via numerical optimization, e.g., by applying a finite-differencing scheme in conjunction with a gradient descent algorithm (see e.g., [215]). Alternatively, one could use Markov Chain Monte Carlo (MCMC) techniques to sample directly from the posterior in order to obtain the log-likelihood landscape [161]. The MCMC approach is particularly advantageous when the scale separation parameter γ is not known *a priori*, as was assumed in the current analysis, since the number of parameter combinations increases exponentially with the number of unknown parameters.

Thus far, we have not considered the effects of measurement noise. In order to obtain the correct parameter likelihoods in the presence of noisy measurements, one would have to marginalize over all possible paths, weighted by the probability of observing the measured values at each point along the sampled path, according to an error model such as normal or log-normal measurement noise. The variance of the noise then constitutes an additional unknown parameter σ which would have to be inferred. Integrating over all possible sample paths is of course computationally intractable due to the enormity of the number of such paths, even if some truncation of the possible path space is made, e.g., by neglecting paths for which the probability of observing the measured data points lies below some arbitrarily small threshold. Alternatively, rather than integrating over all possible paths to obtain the true marginal parameter likelihoods, one could apply a variant of the expectation maximization algorithm [216] in which case the most likely parameter set (a, b, γ, σ) is inferred along with the "true" latent paths $\mathbf{m}$ and $\mathbf{n}$ for mRNA and protein, respectively. A similar approach was employed by Suter, *et al.* [217], wherein they use the zeroth-order model presented in [111] along with simplifying assumptions in order to perform parameter inference from protein time series.

To improve the utility of the uniform propagator for parameter inference, it is necessary to eliminate the non-physical transition probabilities, which can be achieved via the inclusion of higher-order terms in the perturbation parameter ε in the corresponding asymptotic expansion, as the current approximation in Eq. (4.3) is accurate only up to and including first order terms in ε. Alternatively, the "fast" and "slow" propagators that were derived separately in [112], at first order in ε, could be "patched" at some suitable timepoint so that positivity is ensured throughout. Further improvement is likely possible for specific parameter regimes (a, b, γ) in which the relative orders of magnitude of the three parameters naturally suggest a γ-dependent rescaling of a or b. Another possible application of the uniform propagator would be to combine it with other techniques, such as moment equations, in order to perform approximate parameter inference by attempting to match simultaneously the predicted steady-state distributions and autocorrelation

functions of the model to empirical observations. The uniform propagator provides a more exact approximation of the steady-state distribution in the two-stage model for gene expression, as is shown in [112], and is thus potentially well suited to such an approximate inference scheme.

Chapter 5

Inferring gene regulation models using particle filtering

5.1 Introduction

In real world settings, data are often incomplete in the sense that only some of the variables are measured. Due to various constraints, data are often also only obtained at discrete timepoints, leading to uncertainty about unobserved transitions of the system between observations. Moreover, the measurement process is typically imperfect, leading to an associated measurement noise that must be also be taken into consideration when attempting to infer the underlying mechanism and parameters from observed data. In the case of proliferating cell populations, the problem is made more difficult due to the branching nature of the dataset. As cells divide, the population size increases and additional uncertainity is introduced by the unobserved division process for which the initial state of each daughter cells is unknown. Thus, when trying to perform inference using such datasets, it is necessary to utilize sophisticated algorithms tailored to proliferating populations.

In this chapter, I develop an algorithm for parameter inference using discretely and partially observed time series data obtained from proliferating cellular populations. The algorithm is designed to exploit the genealogical structure of the data in order to better estimate the initial conditions of each daughter cell, leading to reduced uncertainty in the parameters of the mechanistic model. I then test the algorithm's performance using synthetic data designed to be qualitatively similar to data produced by time-lapse fluorescence microscopy experiments.

I consider a system consisting of of a single gene, its mRNA and its protein product. Due to the low copy numbers of the system (one promoter, a couple hundred mRNAs), I consider only stochastic models capable of capturing the intrinsic noise arising from random binding and unbinding of the promoter, and stochastic mRNA and protein synthesis and degradation. These chemical species constitute a chemical reaction network and evolve according to the chemical master equation, see Section 2.4.2. A number of modeling strategies exist for approximating the CME since it cannot typically be solved analytically. However, these methods are best suited to monostable systems with relatively large numbers of molecules; in the limit of small molecule numbers, methods such as the

linear noise approximation, or diffusion approximation (see Section 2.4.2) are likely to fail [94]. Another strategy would be to directly solve the CME using methods like the finite state projection and variants (see Section 2.4.2). However, the directly solution becomes computationally infeasible for larger systems containing hundreds of thousands or millions of proteins.

Recently, approximate inference methods based Approximate Bayesian Computing (ABC) have gained attention [218, 219]. ABC has the advantage of being likelihood-free: parameter estimates are obtained purely by filtering sampled parameters on the basis of the discrepancy between simulated and observed data points (or statistics thereof). Thus ABC presents an attractive alternative to more classical methods. One ABC-based algorithm was recently proposed by Loos *et al.* for inferring model parameters using multivariate test statistics based on the population snapshot distributions as a distance metric in the ABC framework [220]. The method was shown to also be suitable for tree-structured data for a one-stage model (only mRNA) with no measurement error. However, ABC suffers the great disadvantage that it uses heuristics to approximate the posterior density of model parameters, making it difficult to assess the correctness of the resulting estimates. Moreover, ABC induces an unknown loss of information, which can lead to incorrect model choice in a model comparison scenario [221]. ABC methods can also be quite sensitive to the choice of tuning parameters, and may not represent a robust estimate of the true posterior distribution. Thus, where possible exact methods are preferred.

The algorithm presented in this chapter utilizes particle filters to infer the unknown regulatory mechanism and associated model parameters underlying discretely-observed protein count time series. Particle filters represent a type of sequential Monte Carlo (SMC) algorithm for generating successive approximations to the posterior joint density of latent trajectories and model parameters [222]. Particle filtering algorithms are similar to previously developed methods, including the Kalman filter and extended Kalman filter, in that they use a model to predict the output of the system at the next time step and subsequently use new observation data to update the predicted state of the system [223, 224]. However, particle filters make no assumptions about the linearity or Gaussianity of system transitions, making them much more suitable for inference and state estimation in complex, non-linear systems. Particle filters are also reminiscent of more recent ABC algorithms. However, unlike ABC, the SMC method utilized by particle filters samples from the *exact* posterior distribution at each iteration, in the limit of infinitely many samples, which leads to better parameter estimates and allows for model comparison.

Particle filters have previously been successfully applied for inference in stochastic systems with non-linear, non-Gaussian transition densities where analytical solutions are generally not possible. Such transition densities are frequently encountered in models of gene expression, e.g. arising due to interactions at the promoter. The particle filter can be directly applied to partially observed, noisy trajectories as long as the statistical distribution of the error is known. Because the particle filter makes no parametric assumptions about the transition density of the state space (proceeding instead via simulation), it is also possible to capture arbitrary, potentially multimodal distributions. Such flexibility is advantageous in the context of gene expression models, where bimodal distributions may arise via slow promoter switching between active and inactive conformations. Thus particle filters present a very attractive option for inferring the latent state space and

associated model parameters from noisy trajectories.

However, although particle filters have previously been applied to inference in gene expression models, see e.g. [138, 139, 225], to my knowledge no attempt has been made to perform inference of stochastic models using *tree-structured* data arising from colonies of proliferating cells, i.e. cellular genealogies. I thus propose an extension to the bootstrap particle filter [226], with explicit modeling of the cell division process and allocation of cellular contents to daughter cells. This additional modeling step improves inference and model selection performance compared to an inference procedure which ignores the genealogical structure of the dataset. The algorithm has the further advantage of fitting each cellular trajectory individually, thus maximizing the amount of information used for inferring model parameters, unlike inference algorithms like ABC which attempt only to fit various statistics of the whole population.

5.2 Mathematical background

5.2.1 Bootstrap particle filter

The objective of the bootstrap particle filter is to sample from the posterior joint density $P(\mathbf{X}, \boldsymbol{\theta}|\mathbf{D}, M)$ of latent trajectories $\mathbf{X}$, and parameters $\boldsymbol{\theta}$ for a model M, given observed data $\mathbf{D}$. The latent trajectories $\mathbf{X}$ are realizations of a stochastic process (see Section 2.3.2), and the data $\mathbf{D}$ represent noisy observations of (a function of) the latent process $\mathbf{X}$ at discrete times. Often the observations are only of a subset of $\mathbf{X}$.

For now we drop the index M for simplicity; when comparing models we will again introduce this notation. The posterior joint density depends on the likelihood $P(\mathbf{D}|\boldsymbol{\theta})$ and prior distribution $\pi(\boldsymbol{\theta})$ according to Bayes' Law (2.2):

$$P(\mathbf{X}, \boldsymbol{\theta}|\mathbf{D}) = \frac{P(\mathbf{D}|\mathbf{X}, \boldsymbol{\theta})P(\mathbf{X}, \boldsymbol{\theta})}{P(\mathbf{D})} = \frac{P(\mathbf{D}|\mathbf{X})P(\mathbf{X}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{P(\mathbf{D})}. \tag{5.1}$$

The simplification on the right side of (5.1) is possible since the probability of observing data $\mathbf{D}$ given a latent trajectory $\mathbf{X}$ depends only on $\mathbf{X}$ and not on the underlying parameters $\boldsymbol{\theta}$ of the stochastic process (measurement error is considered separately). The probability $P(\mathbf{X}|\boldsymbol{\theta})$ captures the evolution of the stochastic processes parametrized by $\boldsymbol{\theta}$.

In general $\mathbf{D}$ constitutes a set of (multivariate) observations of some subset or function of the true latent trajectory $\mathbf{X}$, given by $P(\mathbf{D}|\mathbf{X}) = g(\mathbf{D}|\mathbf{X}, \boldsymbol{\eta})$, where g is the observation function with parameter $\boldsymbol{\eta}$. For example, g might be a Gaussian in which case $\boldsymbol{\eta}$ contains the variance of the measurement process, and potentially a scaling factor. For brevity I omit the dependence on $\boldsymbol{\eta}$ when writing the likelihood function $P(\mathbf{D}|\mathbf{X})$.

The latent trajectory $\mathbf{X}$ represents the configuration of the system in terms of value of each stochastic variable, at each point in time. We restrict ourselves to chemical reaction networks for which the stochastic process is a Markov jump process on the integer lattice corresponding to molecular copy numbers, according to the reactions in the chemical reaction network, see Section 2.4.1. For such a system, the likelihood of the complete latent trajectory $P(\mathbf{X}|\boldsymbol{\theta})$ can be computed [227], and exact samples of $\mathbf{X}|\boldsymbol{\theta}$ can be generated e.g. using Gillespie's algorithm [117], see Section 2.4.3. Note that the transition density (i.e. the probability distribution of $\mathbf{X}$ at a future timepoint) of the stochastic system is in

general not known, but can be approximated for small systems, e.g. using the Finite State Projection (see Section 2.4.2).

Assuming uncorrelated errors in the observation function g, the likelihood $P(\mathbf{D}|\mathbf{X})$ factorizes as:

$$P(\mathbf{D}|\mathbf{X}) = \prod_{i=0}^{N} P(\mathbf{D}_i|\mathbf{X}_i). \tag{5.2}$$

for a series of N observations. The variables $\mathbf{D}_i$ and $\mathbf{X}_i$ indicate the observation and value of the latent state at time t_i, respectively.

Furthermore, the stochastic process $\mathbf{X}$ is Markovian (see Section 2.3.2), thus the likelihood of the trajectory decomposes as:

$$P(\mathbf{X}|\boldsymbol{\theta}) = P(\mathbf{X}_0) \prod_{i=1}^{N} P(\mathbf{X}_{[t_{i-1},t_i]}|\mathbf{X}_{i-1}, \boldsymbol{\theta}) \tag{5.3}$$

where $\mathbf{X}_{[t_{i-1},t_i]}$ indicates the full path of the stochastic process in the time interval $[t_{i-1}, t_i]$. The variable $\mathbf{X}$ without subscripts is used as shorthand for the entire trajectory, i.e. $\mathbf{X} = \mathbf{X}_{[t_0,t_N]}$.

Following the derivation of Gordon *et al.* [226], we combine (5.2) and (5.3), and substitute into (5.1), to obtain a new expression for the posterior density:

$$\begin{aligned}
P(\mathbf{X}, \boldsymbol{\theta}|\mathbf{D}) &= \prod_{i=1}^{N} \frac{P(\mathbf{D}_i|\mathbf{X}_i)P(\mathbf{X}_{[t_{i-1},t_i]}|\mathbf{X}_{i-1}, \boldsymbol{\theta})}{P(\mathbf{D}_i)} P(\mathbf{X}_0|\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \\
&= \frac{P(\mathbf{D}_N|\mathbf{X}_N)P(\mathbf{X}_{[t_{N-1},t_N]}|\mathbf{X}_{N-1}, \boldsymbol{\theta})}{P(\mathbf{D}_N)} \prod_{i=1}^{N-1} \frac{P(\mathbf{D}_i|X_i)P(\mathbf{X}_{[t_{i-1},t_i]}|\mathbf{X}_{i-1}, \boldsymbol{\theta})}{P(\mathbf{D}_i)} P(\mathbf{X}_0|\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \\
&= \frac{P(\mathbf{D}_N|\mathbf{X}_N)P(\mathbf{X}_{[t_{N-1},t_N]}|\mathbf{X}_{N-1}, \boldsymbol{\theta})}{P(\mathbf{D}_N)} P(\mathbf{X}_{[t_0,t_{N-1}]}, \boldsymbol{\theta}|\mathbf{D}_0, \ldots, \mathbf{D}_{N-1})
\end{aligned} \tag{5.4}$$

This can be rewritten as

$$P(\mathbf{X}, \boldsymbol{\theta}|\mathbf{D}_{0:N}) = w \frac{P(\mathbf{X}_{[t_{N-1},t_N]}|\mathbf{X}_{N-1}, \boldsymbol{\theta})}{P(\mathbf{D}_N)} P(\mathbf{X}_{[t_0,t_{N-1}]}, \boldsymbol{\theta}|\mathbf{D}_{0:N-1}) \tag{5.5}$$

where $w = P(\mathbf{D}_N|\mathbf{X}_N)$ and $\mathbf{D}_{0:K} = (\mathbf{D}_0, \ldots, \mathbf{D}_K)$.

Hence, there is a simple update rule relating the posterior distribution using observations until timepoint t_{N-1} to the posterior distribution with the next observation at timepoint t_N. Note also that one can generate a sample from the posterior at time t_i, $P(\mathbf{X}_{[t_0,t_i]}, \boldsymbol{\theta}|\mathbf{D}_{0:i})$, by first sampling a trajectory from the marginal distibution $P(\mathbf{X}_{[t_0,t_i]}|\mathbf{D}_{0:i})$ and then sampling a parameter $\boldsymbol{\theta}|\mathbf{X}_{[t_0,t_i]}$, suggesting a Gibbs sampling approach.

These observations and the recursive factorization of the joint posterior (5.5) motivates the so-called bootstrap particle filter [226]:

Algorithm 1: Bootstrap particle filter

Data: A set of observed data points $\mathbf{D} = (\mathbf{D}_0, \dots, \mathbf{D}_N)$ at timepoints $(t_0, \dots, t_N)$, parameter prior $\pi : \mathbb{R}^d \to \mathbb{R}_{0,+}$, observation function $g(\mathbf{D}|\mathbf{X}, \boldsymbol{\eta}) = P(\mathbf{D}|\mathbf{X})$, number of particles K

Result: A set of particles $\left\{(\mathbf{X}^{(k)}, \boldsymbol{\theta}^{(k)})\right\}_{k=1}^{K}$ sampled from the posterior density $P(\mathbf{X}, \boldsymbol{\theta}|\mathbf{D})$

initialization;
for *k=1...K* **do**
    Sample parameter values from the prior: $\boldsymbol{\theta}^{(k)} \sim \pi(\boldsymbol{\theta})$;
    Sample initial state conditional on first observed data point: $\mathbf{X}_0^{(k)} \sim g^{-1}(\cdot|\mathbf{D}_0, \boldsymbol{\eta})$;
    Initialize particle weight to $w_0^{(k)} := 1/K$
main loop ;
for *i=1...N* **do**
    Generate a set of particle indices $\epsilon^{(k)} \in \{1, \dots, K\}, k = 1, \dots, K$ such that $P(\epsilon^{(k)} = a) = w_{i-1}^{(a)} / \sum_{\ell=1}^{K} w_{i-1}^{(\ell)}$;
    for $k = 1 \dots K$ **do**
        Generate a sample trajectory $\mathbf{X}_{[t_{i-1},t_i]}^{(k)} \sim P(\cdot|\mathbf{X}_{i-1}^{(\epsilon^{(k)})}, \boldsymbol{\theta}^{(\epsilon^{(k)})})$;
        Concatenate to previously sampled trajectory: $\mathbf{X}_{[t_0,t_i]}^{(k)} := [\mathbf{X}_{[t_0,t_{i-1}]}^{(\epsilon^{(k)})}, \mathbf{X}_{[t_{i-1},t_i]}^{(k)}]$;
        Set the weight of the k^{th} particle to the likelihood: $w_i^{(k)} := P(\mathbf{D}_i|\mathbf{X}_i^{(k)}) = g(\mathbf{D}_i|\mathbf{X}_i^{(k)}, \boldsymbol{\eta})$;
        Generate a new set of parameters $\boldsymbol{\theta}^{(k)}$ from the conditional density: $\boldsymbol{\theta}^{(k)} \sim P(\boldsymbol{\theta}|\mathbf{X}_{[t_0,t_i]}^{(k)})$
Sample from the posterior ;
Generate a set of particle indices $\epsilon^{(k)} \in \{1, \dots, K\}, k = 1, \dots, K$ such that $P(\epsilon^{(k)} = a) = w_N^{(a)} / \sum_{\ell=1}^{K} w_N^{(\ell)}$;
Construct a sample of K particles from the posterior: $\left\{(\boldsymbol{\theta}^{(\epsilon^{(k)})}, \mathbf{X}^{(\epsilon^{(k)})})\right\}_{k=1}^{K}$

The recursive particle filter begins by sampling parameters $\boldsymbol{\theta}$ from the prior $\pi(\boldsymbol{\theta})$, and an initial condition for the state $\mathbf{X}$, for an ensemble of K *particles*, i.e. each particle is a sample from the joint density of $\mathbf{X}$ and $\boldsymbol{\theta}$. If the observation function $g(\mathbf{D}|\mathbf{X}, \boldsymbol{\eta})$ is invertible, then $\mathbf{X}$ can be sampled from the inverse distribution; otherwise, $\mathbf{X}$ can be chosen arbitrarily in a way that is consistent with prior knowledge.

At each iteration i, the particles are resampled according to their normalized weights $w_i^{(k)} / \sum_{\ell=1}^{K} w_i^{(\ell)}$, such that particles that have a state $\mathbf{X}_i^{(k)}$ for which the current observation $\mathbf{D}_i$ is likely are sampled more frequently. Resampling according to the normalized weights generates a collection of samples with empirical distribution which converges to the true posterior distribution $P(\mathbf{X}_{[t_0,t_N]}|\mathbf{D})$ asymptotically with the number of samples (see [228] for proof); the method is illustrated in Figure 5.1.

Using the latent states of the resampled particles as initial conditions and parameters sampled from their conditional distributions, the particles are propagated to the next timepoint by sampling from the stochastic propagator. In the case of chemical reaction networks, this can be achieved simply by using the stochastic simulation algorithm or variants (see Section 2.4.3). The particles are then reweighted according to the likelihood of the next observation. In this Gibbs sampling approach, first a particle is sampled according to its weight, then a parameter $\boldsymbol{\theta}$ is sampled conditional on the particle's latent trajectory $\mathbf{X}$ up to the current timepoint. The result of the recursive particle filter is an exact sample from posterior joint density of $(\mathbf{X}, \boldsymbol{\theta}|\mathbf{D})$, as formulated in (5.4).

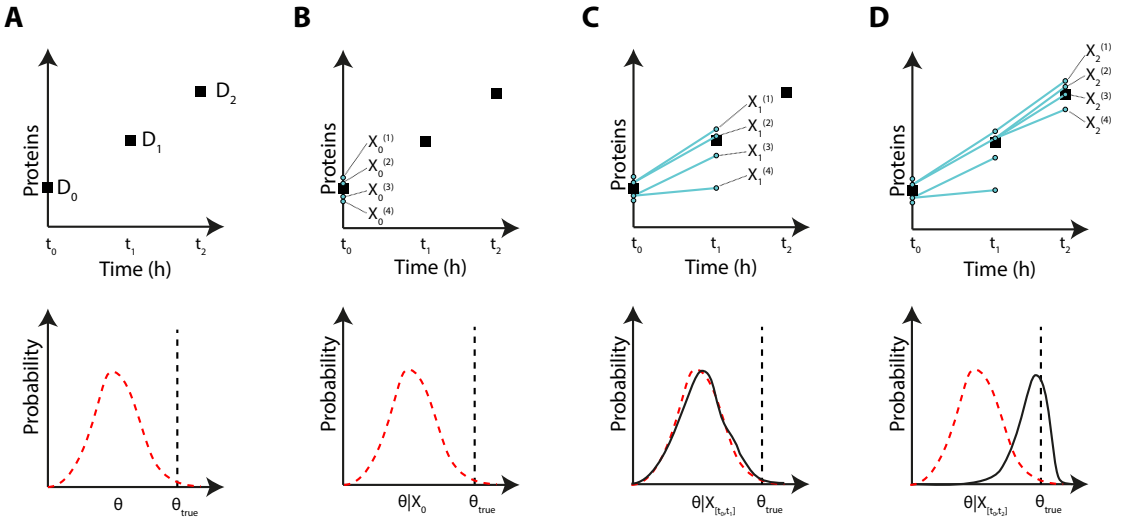

Figure 5.1: Illustration of the bootstrap particle filter. A. The bootstrap particle filter requires a series of observation $\mathbf{D} = (\mathbf{D}_0, \mathbf{D}_1, \ldots)$ (top) and a prior distribution for model parameters $\pi(\boldsymbol{\theta})$ (bottom) as input. B. Particles are initialized by sampling latent states $\mathbf{X}_0^{(k)}$ for each particle k. Parameters $\boldsymbol{\theta}$ are sampled from the prior distribution $\pi(\boldsymbol{\theta})$. C. The latent trajectories are resampled according to the likelihood $w_0^{(k)} = P(\mathbf{D}_0|\mathbf{X}_0^{(k)})$ and propagated to the next time step using stochastic simulations to generate new states $\mathbf{X}_1^{(k)}$ at timepoint t_1. Model parameters for each latent trajectory are resampled from the conditional distribution $P(\boldsymbol{\theta}|\mathbf{X}_{[t_0,t_1]}^{(k)})$. D. At each iteration, the weights are recomputed and the particles are resampled. Resampled particles are propagated to the next timepoint and the parameters are resampled conditional on the resampled latent trajectories. Over time the posterior parameter distribution converges to the true value.

5.2.2 Gamma priors

The particle filter is significantly simplified if one assumes a gamma prior distribution (see Section A.5) for each model parameter, and conditional independence between the model parameters:

$$\pi(\boldsymbol{\theta}) = \prod_{i=1}^{d} \pi_i(\boldsymbol{\theta}_i) = \prod_{i=1}^{d} \mathrm{Ga}(\theta_i; \alpha_i, \beta_i) \tag{5.6}$$

where α_i and β_i are the hyperparameters for the distribution of $\boldsymbol{\theta}_i, i = 1 \dots d$. The conditional independence between model parameters is often justified as information about the covariance of biological constants is usually not available. Note that the gamma distribution is flexible enough to take on a variety of shapes ranging from exponentially-distributed to peaked, with tunable skewness via the choice of the parameters.

Using gamma priors for each model parameter, the conditional probability $P(\mathbf{X}|\boldsymbol{\theta})$ of a particular realization of the Markov jump process is conjugate to the prior, such that the conditional density $P(\boldsymbol{\theta}|\mathbf{X})$ is also gamma-distributed, see Wilkinson *et al.* [227], p. 281:

$$\begin{aligned} P(\boldsymbol{\theta}|\mathbf{X}) &= \frac{P(\mathbf{X}|\boldsymbol{\theta})}{P(\mathbf{X})}\pi(\boldsymbol{\theta}) = \frac{P(\mathbf{X}|\boldsymbol{\theta})}{P(\mathbf{X})} \prod_{p=1}^{d} \mathrm{Ga}(\theta_p; \alpha_p, \beta_p) \\ &= \prod_{p=1}^{d} \mathrm{Ga}(\theta_p; \alpha_p + r_p, \beta_p + G_p) \end{aligned} \tag{5.7}$$

where r_i is the number of reaction firings of reaction i in the process $\mathbf{X}$, and G_p is proportional to the integrated propensity ($a_p(\mathbf{X})$) of reaction p as given in Table (2.1): $G_p = \frac{1}{k_p} \int_0^T a_p(\mathbf{X}(s))ds$. Here k_p is the reaction constant of reaction p, which cancels the same term in the propensity function a_p, thus rendering G_p dependent only on the instantaneous configuration of the system at all points along the trajectory, and not on the reaction constants. Hence, a new sample for $\boldsymbol{\theta}$ given the newly simulated trajectory (line 13 of Algorithm 1) can be generated by simply sampling from the updated gamma posterior (5.7); furthermore, the summary statistics r_p and G_p are sufficient for describing the posterior distribution of θ_p, thus the full trajectories $\mathbf{X}$ do not need to be stored, leading to reduced memory consumption.

5.2.3 Model comparison

The particle filtering approach presented above can also be used for performing model comparison via Bayes Factors, i.e. by computing $\frac{P(M_1|\mathbf{D})}{P(M_2|\mathbf{D})}$, the ratio of the posterior probabilities of each model 1 (M_1) to model 2 (M_2). Using Bayes' law, one can reformulate the marginal probability of model M as:

$$P(M|\mathbf{D}) = \frac{P(\mathbf{D}|M)P(M)}{P(\mathbf{D})} \tag{5.8}$$

Following Wilkinson *et al.* [227], p. 294, we can approximate the marginal likelihood of the model $P(\mathbf{D}|M)$ using the sampled particles at each iteration i. Firstly, the distribution of the observed data at time t_{i+1} depends only observations up to t_i: $P(\mathbf{D}_{i+1}|\mathbf{D}, M) =$

$P(\mathbf{D}_{i+1}|\mathbf{D}_{0:i}, M)$. Moreover, this probability is approximated by the expectation of the likelihood, or weights $w^{(k)}$, of the particles:

$$\begin{aligned} P(\mathbf{D}_{i+1}|\mathbf{D}_{0:i}, M) &= \int P(\mathbf{D}_{i+1}|\mathbf{X}_{i+1})P(\mathbf{X}_{i+1}|\mathbf{D}_{0:i}, M)d\mathbf{X}_{i+1} \\ &\approx \frac{1}{K}\sum_{k=1}^{K} \underbrace{P(\mathbf{D}_{i+1}|\mathbf{X}_{i+1}^{(k)})}_{w_{i+1}^{(k)}} \end{aligned} \tag{5.9}$$

where the $\mathbf{X}_{i+1}^{(k)}$ are sampled (via the particle filter) from the marginal posterior up to time t_{i+1} given by $P(\mathbf{X}_{i+1}|\mathbf{D}_{0:i})$. This is nothing more than a Monte Carlo approximation of the integral, which provides an unbiased approximation of $P(\mathbf{D}_{i+1}|\mathbf{D}_{0:i}, M)$ with variance decreasing as K^{-1} [222].

Next, since the distribution of each observation depends only on previous observations, the marginal probability of the entire set of observations $P(\mathbf{D}|M)$ is given by the product:

$$P(\mathbf{D}|M) = P(\mathbf{D}_0)\prod_{i=1}^{N} P(\mathbf{D}_{i+1}|\mathbf{D}_{0:i}, M). \tag{5.10}$$

Assuming *a priori* equally likely models, the factor of $P(M)$ in (5.8) cancels between the two models and the Bayes Factor reduces to the ratio of marginal likelihoods. Although it may also be possible to use other Monte Carlo methods for approximating the marginal log likelihood, e.g. thermodynamic integration [229], that would necessitate running several more iterations of the inference procedure to obtain samples from the posterior with different power-likelihoods, leading to additional computational overhead.

5.3 Implementation

I implemented a highly parallelized version of the particle filtering algorithm (Algorithm 1) in Matlab, with a custom implementation of the forward simulation algorithm for generating new latent trajectories $\mathbf{X}$ using the Gillespie algorithm (SSA)[117]. The forward simulation code was implemented using vectorized operations for computing the (exponentially distributed) waiting times of each reaction for blocks of 2×10^4 simulations simultaneously. Blocks of simulations were distributed to different cores of a multicore machine for efficient parallel sampling from the stochastic process. Simulation code was further optimized by automatically converting to `C` code (using the codegen toolbox) and compiling using the matlab compiler `mex`.

For each simulation the necessary statistics are stored, including the number of reaction firings r_j, and the integral of the (scaled) propensity functions over the simulated trajectory, G_j, for each reaction j. The storage space required grows linearly with the number of particles, latent species and parameters. The number of simulations necessary grows linearly with the number of timepoints simulated and the number of reactions, since for each reaction new random numbers must be sampled for reaction waiting times, although it may be less for more efficient implementations of the forward simulation algorithm. In some cases the number of proteins simulated is very large, in which case I use an

approximation to facilitate the stochastic simulation: I perform the full SSA for the DNA and mRNA-related reactions, but approximate the number of birth and death reactions for protein over the interval where DNA and mRNA copy numbers are constant, using the τ-leaping approximation [122], i.e. the number of firings for protein-related reactions over this interval is Poisson-distributed (see Section A.4). This approximation provides considerable speedup making the simulation possible even when the number of simulated molecules exceeds 10^6. The time interval for the Poisson approximation of the τ-leaping variant was reduced such that the expected change in the reaction propensities was at most 1% of the present value.

5.4 Application to simulated data

Using the particle filter with gamma priors for each model parameter, I performed a series of tests to determine whether this inference framework is suitable for inferring model parameters and identifying the best model for Nanog autoregulation. To this end, I constructed several simulated data sets that resemble actual time-lapse fluorescence microscopy experiments. In each dataset, proteins copy numbers are observed (with some measurement error) at regular intervals, whereas the state of the promoter and the copy number of mRNA are unknown throughout.

5.4.1 Models investigated

I generated synthetic data from three models featuring either no feedback, positive or negative feedback. Each model is a three-stage model, with active DNA (D^*), inactive DNA (D), mRNA (M), and protein (P); analytical approximations to such models have been the subject of much recent work (see e.g. [109, 112, 230, 231] and Chapter 4). I model only a single promoter, i.e. the total copy number of active and inactive DNA is always 1. In the case of the positive feedback, protein increases the rate of DNA activation. Moreover, the DNA activation rate depends quadratically on the protein number P, as would be the case if the protein first dimerized, assuming fast reversible dimerization of the protein—then the quantity of dimer is proportional to the square of P where the additional constant is absorbed into k_{on}, the activation rate of the DNA. In the case of negative feedback, the rate of deactivation depends on the square of the protein number, following similar reasoning. Hence the models only differ in the activation and inactivation rate of the DNA, and are summarized in Table 5.1. All simulated models only include transcriptional regulation, although it is possible to extend the simulations to post-transcriptional regulation. In each case, only the protein is observed, at discrete timepoints.

	Propensity Function		
Reaction	+ Feedback	- Feedback	$\emptyset$ Feedback
$DNA \rightarrow DNA^*$	$k_{\text{on}}\ D\ P^2$	$k_{\text{on}}\ D$	$k_{\text{on}}\ D$
$DNA^* \rightarrow DNA$	$k_{\text{off}}\ D^*$	$k_{\text{off}}\ D^*\ P^2$	$k_{\text{off}}\ D^*$
$DNA^* \rightarrow DNA^* + mRNA$	$k_m\ D^*$	$k_m\ D^*$	$k_m\ D^*$
$mRNA \rightarrow \emptyset$	$d_m\ M$	$d_m\ M$	$d_m\ M$
$mRNA \rightarrow mRNA + Protein$	$k_p\ M$	$k_p\ M$	$k_p\ M$
$Protein \rightarrow \emptyset$	$d_p\ P$	$d_p\ P$	$d_p\ P$

Table 5.1: Autoregulation models used for model selection. The Positive (+) Feedback model has DNA activation propensity that increases quadratically with the number of proteins. The Negative (-) Feedback model has DNA deactivation propensity that increases quadratically with the number of proteins. Species are abbreviated as inactive DNA (D), active DNA (D^*), mRNA (M) and Protein (P).

5.4.2 Tree simulations

For each model I generated a synthetic tree-structured dataset. Specifically, a single progenitor cell was simulated from the initial condition with DNA inactive, 10 molecules of mRNA present, and a random amount of protein uniformly sampled from $[2000, 2500]$. The cell was simulated for a random lifetime with uniform distribution on $[8.5, 11.5]$ hours at which time it gives rise to two daughter cells, representing a cellular division event. It is assumed that each molecule of the mother cells contents is allocated to the two daughter cells with equal probability, giving rise to a binomial distribution.

At the time of division, two daughter cells are created from the mother cell, such that:

- The DNA state (active or inactive) of the mother cell is maintained by the daughter cells
- mRNA is partitioned binomially ($p = 0.5$) to the two daughter cells (such that the total mRNA is conserved)
- Protein is partitioned binomially ($p = 0.5$), conserving the total protein copy number P of the mother cell. Since the P is generally very large, I approximate the binomial distribution using a normal distribution with mean $\mu = 0.5P$ and variance $\sigma^2 = 0.5(1 - 0.5) = 0.25P$, which provides a computationally efficient approximation to the binomial distribution for large values of the argument.

The simulation was continued in the same way until three generations had completed, with cell lifetimes sampled from the same distribution. The protein quantity was output at time intervals of 0.5 time units, and Gaussian measurement error with standard deviation 200 was added to each observation. The results of the partially observed, noisy, tree-based simulations are shown in Figure 5.2.

5.4.3 Choice of prior and model parameters

For the "No Feedback" ($\emptyset$) model, parameters were chosen so as to be reasonably consistent with the dynamics of mESCs. For example, DNA activation and inactivation rates are

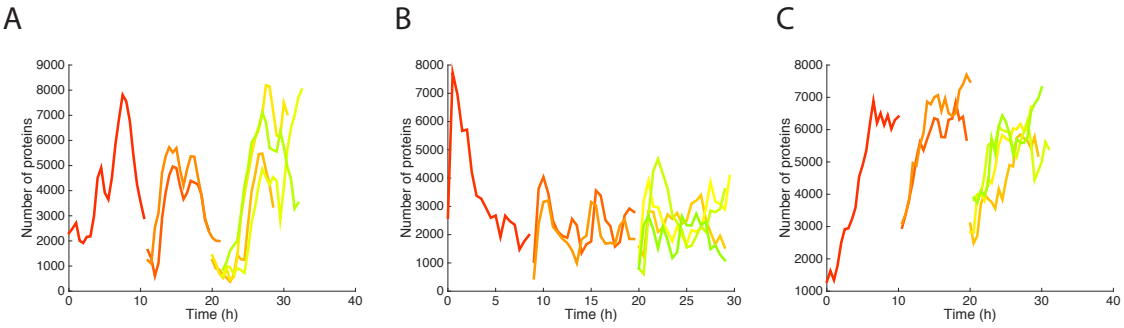

Figure 5.2: Simulated colonies for A. No Feedback, B. Negative Feedback, and C. Positive Feedback models. Individual cells are shown using different colors for ease of visualization.

Parameter	Description	Units	Model		
			+ Feedback	- Feedback	∅ Feedback
k_{on}	DNA activation	h^{-1}	$5 \times 10^{-7}/P^2$	10	1
k_{off}	DNA inactivation	h^{-1}	1	$1 \times 10^{-4}/P^2$	1
k_m	transcription	h^{-1}	40	100	40
d_m	mRNA degradation	$h^{-1}\text{mRNA}^{-1}$	3	3	3
k_p	translation	$h^{-1}\text{mRNA}^{-1}$	200	2000	300
d_p	protein degradation	$h^{-1}\text{Protein}^{-1}$	0.4	0.4	0.4

Table 5.2: Parameters used for simulation of each model.

chosen so that the DNA changes state on the order of hours, leading to relatively long periods of quiescence or active transcription, as reported e.g. in [147]. Protein degradation rates were chosen to be of the same order of magnitude as the measured half-lives of Nanog [154], about 5 hours; production rates were chosen such that maximally a few hundred mRNA molecules are produced. Proteins were on the order of tens of thousands, which is less than found in mESCs. Nonetheless the tests were performed with relatively small quantities of proteins in order to facilitate the stochastic simulation, which slows with increasing numbers of molecules. All model parameters are listed in Table 5.2.

For the "Positive Feedback" (+) and "Negative Feedback" (-) models, parameters were chosen in a similar way, so as to produce reasonable quantities of proteins and mRNAs. DNA activation and inactivation rates were chosen so that the expected waiting time of DNA transitions was on the order of hours.

			+ Feedback		- Feedback		∅ Feedback	
Parameter	Description	Units	α	β	α	β	α	β
k_{on}	DNA activation	h^{-1}†	10	1.2×10^7	5	0.8	5	2
k_{off}	DNA inactivation	h^{-1}‡	14	20	7	10^5	5	2
k_m	transcription	h^{-1}	12	0.2	12	0.2	12	0.2
d_m	mRNA degradation	$h^{-1}\text{mRNA}^{-1}$	7	1	7	1	7	1
k_p	translation	$h^{-1}\text{mRNA}^{-1}$	4.3	0.005	4.3	0.005	4.3	0.005
d_p	Protein degradation	$h^{-1}\text{Protein}^{-1}$	5	5	5	5	5	5

Table 5.3: Parameters of the gamma priors used for inference in the three autoregulatory models. † For the Positive (+) Feedback model, the DNA activation rate has units of $h^{-1}\text{Protein}^{-2}$. ‡ For the Negative Feedback model, the DNA inactivation rate has units of $h^{-1}\text{Protein}^{-2}$.

5.4.4 Parameter inference on single cells

I first tested the inference procedure by applying Algorithm 1 to each individual cell of each of the simulated data sets shown in Figure 5.2. The inference algorithm was performed using each of the three models, against each cellular trajectory, with three replicates per combination, yielding a total of 189 inference combinations. In each case 10^5 particles were used for inference, as preliminary testing suggested this was sufficient to prevent degeneracy of samples in the inference algorithm, i.e. adequately many simulated latent trajectories were resampled at each iteration and propagated to the next iteration of the algorithm. However, even with this fairly large number of particles, inference proceeds quickly (within minutes) because of the efficient implementation, compilation to C, and parallelization. Due to the structural differences in the propensity functions of the DNA activation and inactivation between the different models, the scale of the model parameters (such as k_{on} and k_{off}) may vary considerably. Thus, while fitting each of the models to the simulated data, I used priors appropriate to each model. This means that model parameters for DNA activation and inactivation are not directly comparable across models, since the structure of the reactions differ. The priors for the remaining parameters (k_m, d_m, k_p, d_p), however, are identical for each assumed model, see Table 5.3.

For simplicity, I initialize the DNA state of each latent trajectory to either active or inactive with equal probability, and sample the mRNA copy number from a uniform distribution on $[0, 50]$. Proteins are sampled from a Gaussian distribution centered about the first observation, and with standard deviation corresponding to the assumed measurement error.

No Feedback model

I first performed inference on the dataset generated using the No Feedback model (Figure 5.2A) while assuming the correct No Feedback model for inference. I performed three

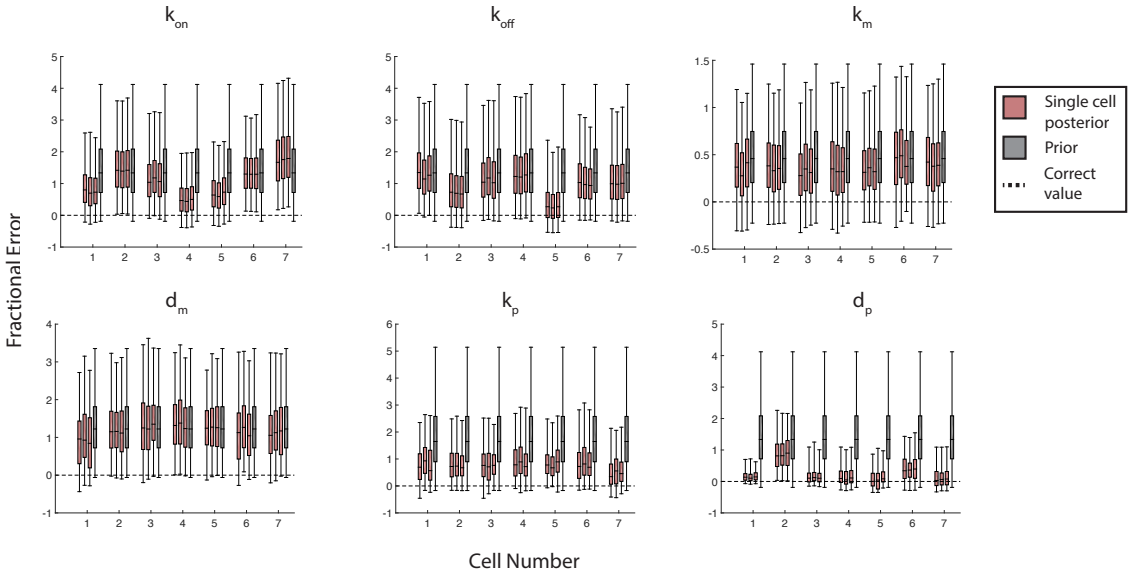

Figure 5.3: Posterior distributions of each replicate of the inference algorithm for data simulated from the No Feedback model, shown as fractional errors of the true value. Each of the 7 cells were fit 3 times independently (red boxes). The quantiles of the prior distribution are shown in gray.

replicates of the inference procedure for each of the seven cells. For simplicity, I report each parameter sample θ using the fractional error, defined as $(\theta - \theta_{\text{true}})/\theta_{\text{true}}$, where θ_{true} is the true value of the model parameter (see Table 5.2). The results are summarized using box-and-whiskers plots showing the interquartile range (box) and extending to contain 95% of the samples (whiskers), see Figure 5.3.

For k_{off}, d_m, k_p and d_p, the single-cell-based inference shows some convergence towards the correct model parameters (i.e. towards zero fractional error) as compared to the prior distributions (shown in gray). For any given cell, the sampled parameters are very consistent (compare the red box-and-whisker plots in Figure 5.3). However the results vary substantially from cell to cell, especially for k_{on} and k_{off}. The protein degradation rate d_p seems to be well inferred for almost every cell. In contrast, the mRNA degradation rate d_m shows little deviation from the prior suggesting that the algorithm is able to extract only little information regarding this parameter from the single-cell trajectories when considered individually.

Next, I fit the same simulated dataset using the other two (incorrect) models and compute the resulting estimate to the marginal likelihood of each model using (5.8). Interestingly, I find that the inference algorithm estimates a significantly higher marginal likelihood for the correct (No Feedback) model compared to the Negative Feedback and Positive Feedback models for 6 out of 7 cells, see Table 5.4; cell 5 weakly favored the Positive Feedback model. In all cases, the No Feedback model was strongly preferred over the Negative Feedback model as seen by the large value of the log Bayes Factors.

Cell	No Feedback	Negative Feedback	Positive Feedback
1	0	-9.3054 (0.2582)	-10.7013 (0.1581)
2	0	-13.5696 (3.3602)	-5.935 (0.0862)
3	0	-12.2594 (1.1507)	-7.3764 (0.1044)
4	0	-15.0681 (1.3747)	-9.8305 (0.0782)
5	-1.1334 (0.121)	-33.2052 (6.1942)	0
6	0	-22.3497 (2.7166)	-6.9995 (0.141)
7	0	-15.0969 (0.4353)	-5.1588 (0.1668)

Table 5.4: Log Bayes Factors for single-cell-based inference on the No Feedback dataset, relative to the best model for each cell. Mean (standard deviation) from three replicates.

Negative Feedback model

Next, I applied the inference algorithm assuming the (correct) Negative Feedback model for each of the 7 cells simulated with the Negative Feedback model shown in Figure 5.2B. As for the No Feedback example, I find that the inference procedure also produces fairly consistent estimates of model parameters across replicates for most cells, see Figure 5.4. However some model parameters, such as d_m, show variable distributions across cells. The model parameters d_m and d_p seem to be inferred well for nearly all cells, while the remaining parameters either show partial convergence to the correct parameters (k_{on}, k_m, kp) or a slight bias in the case of k_{off}.

Computing the Bayes Factors for each model for this dataset, however, I find that the the single-cell based inference favors the incorrect No Feedback model for 3 out of 7 cells, although for one cell the Bayes Factor is too small to be decisive, see Table 5.5. The misclassification of these 3 cells is likely because their trajectories are similar enough to those typical of the No Feedback model that the correct model is not more likely given the observed data. In addition, if the Negative Feedback model is more sensitive to model parameters, then it is possible that the likelihood of each transition is on the average lower for the Negative Feedback model for particles with parameter values that are not very close to the true value, thus leading to a somewhat lower average transition probability and finally to a lower marginal likelihood of the Negative Feedback model for these cells.

Cell	No Feedback	Negative Feedback	Positive Feedback
1	-4.131 (0.3035)	0	-26.3066 (0.8125)
2	-2.502 (0.014)	0	-14.6629 (0.1774)
3	-1.3918 (0.0288)	0	-11.957 (0.1156)
4	0	-4.1386 (0.5486)	-1.0796 (0.0182)
5	0	-1.0327 (0.0177)	-8.6714 (0.1402)
6	0	-5.2632 (2.1421)	-5.0105 (0.0225)
7	-0.2432 (0.031)	0	-8.7574 (0.1)

Table 5.5: Log Bayes Factors for single-cell based inference on the Negative Feedback dataset, relative to the best model for each cell. Mean (standard deviation) from three replicates.

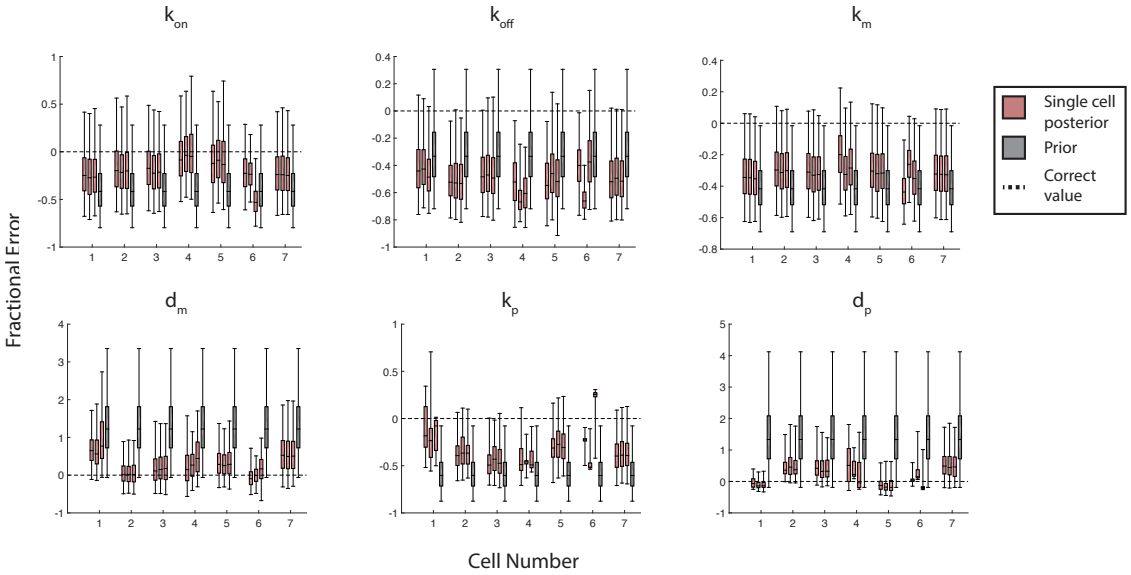

Figure 5.4: Posterior distributions of each replicate of the inference algorithm for data simulated from the Negative Feedback model, shown as fractional errors of the true value. Each of the 7 cells were fit 3 times independently (red boxes). The quantiles of the prior distribution are shown in gray.

Positive Feedback model

Finally, I tested the performance of the algorithm for the Positive Feedback dataset (assuming the correct model), shown in Figure 5.2C. I found that k_p and especially d_p seem to exhibit convergence to the true value, in the latter case showing a peak exactly on the true value for some cells, see Figure 5.5. However, $k_{\text{on}}, k_{\text{off}}$ and k_m show only a small difference from their respective priors, while d_m shows a weak bias away from the true value. Thus for this model it is not possible to infer most model parameters using only individual cells. I note however that the parameter sample distributions generated by the algorithm for each cell are quite consistent over each of the replicates.

Computing the Bayes Factor of the correct model compared to the No Feedback and Negative Feedback models, I find that this single-cell based analysis shows preference for the correct model for all cells, although only weakly so for cells 3 and 7, see Table 5.6.

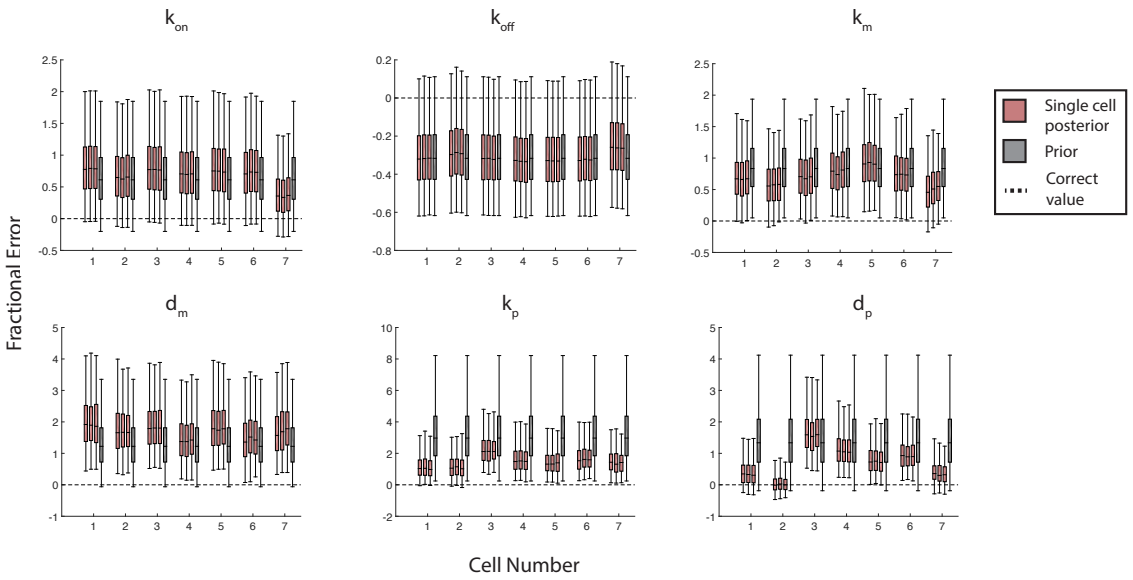

Figure 5.5: Posterior distributions of each replicate of the inference algorithm for data simulated from the Positive Feedback model, shown as fractional errors of the true value. Each of the 7 cells were fit 3 times independently (red boxes). The quantiles of the prior distribution are shown in gray.

Cell	No Feedback	Negative Feedback	Positive Feedback
1	-2.9141 (0.079)	-12.5306 (0.4929)	0
2	-2.901 (0.243)	-14.1889 (1.3588)	0
3	-0.9532 (0.0783)	-9.3542 (0.6612)	0
4	-4.4171 (0.0225)	-18.2459 (2.9465)	0
5	-4.4039 (0.035)	-17.2991 (1.2285)	0
6	-4.2933 (0.043)	-17.5176 (1.3442)	0
7	-0.521 (0.0497)	-11.3935 (0.9244)	0

Table 5.6: Log Bayes Factors for single-cell based inference on the Positive Feedback dataset, relative to the best model for each cell. Mean (standard deviation) from three replicates.

5.4.5 Parameter inference on genealogies

Until now, all inference has been performed using single cells individually, as would be the case if we did not know the genealogical structure of the colony. In this case, it's reasonable to perform inference for each cell separately and compare the inference results. However, since we have the complete genealogical record including all cell divisions, it is possible to exploit the tree structure to improve inference results. This allows us to structure more sensibly initialize the inference algorithm for daughter cells at the moment of division by sampling from the distribution of DNA, mRNA and protein to the daughter cells from the mother cell. This is more informative than assuming arbitrary distributions for the initial conditions, as was previously done for the single-cell-based inference. Secondly, by utilizing the tree structure, the inference algorithm attempts to identify sets of parameter values that generate trajectories that have a high likelihood for multiple cells simultaneously, as the tree branches, i.e. as the cells proliferate. Although it is possible to attempt to perform inference on all cells from the tree simultaneously while neglecting the chronological and genealogical order, numerical experiments revealed this approach to be very inefficient, leading to incorrect convergence of the model parameters due to the great difficulty of fitting many data points simultaneously without informative distributions for latent trajectories and model parameters, as are furnished by the bootstrap particle filter.

The tree-based inference proceeds as described in Algorithm 2. For each cell $j = 1 \dots N$ of the tree, there is a series of N_j observations (i.e. protein measurements) $({}^j\mathbf{D}_1, \dots, {}^j\mathbf{D}_{N_j})$ obtained at times $({}^jt_1, \dots, {}^jt_{N_j})$. A set of K particles is then initialized, where each particle contains both a latent state ${}^1\mathbf{X}_0^{(k)}$ and a set of parameters $\boldsymbol{\theta}^{(k)}$. All particles are initially equalled weighted. The algorithm then iterates through all of the measurement time points, where for simplicity we assume that all measurements are obtained at regular intervals of Δt; however, the method is equally valid for irregular measurement time intervals. At each iteration i, particles are resampled with frequencies proportional to their weights, and cells that are alive at the current timepoint $i\Delta t$ and still living (i.e. observed) at the next timepoint are simulated one time step using the stochastic process conditional on the sampled parameters $\boldsymbol{\theta}^{(k)}$ to generate a sample ${}^j\mathbf{X}^{(k)}_{[i\Delta t,(i+1)\Delta t]}$ of the latent process over the time interval $[i\Delta t, (i+1)\Delta t]$ for cell j. Cells that are alive at the current timepoint, but not at the subsequent timepoint, divide to produce two daughter cells with latent states ${}^{2j}\mathbf{X}^{(k)}_{i+1}$ and ${}^{2j+1}\mathbf{X}^{(k)}_{i+1}$, where the cells are numbered such that cell j gives rise to cells $2j$ and $2j+1$. The joint density of the two daughter cells at time $(i+1)\Delta t$ derives from the same division process previously described for the tree simulations (see Section 5.4.2), and is given by:

$$P(D_1 = d_1, M_1 = m_1, P_1 = p_1, D_2 = d_2, M_2 = m_2, P_2 = p_2 | D_0 = d_0, M_0 = m_0, P_0 = p_0) = \\ \delta(d_1 - d_0)\delta(d_2 - d_0) C^{m_0}_{m_1} 0.5^{m_0} \delta(m_0 - m_1 - m_2) C^{p_0}_{p_1} 0.5^{p_0} \delta(p_0 - p_1 - p_2) \tag{5.11}$$

where $D_j, M_j, P_j, j = 1, 2$ represent the DNA, mRNA and protein states, respectively, of the two daughter cells, and D_0 etc. that of the mother cell; C^n_k denotes the binomial coefficient and $\delta(x)$ the Dirac delta function. After each forward simulation or division step, the likelihood $P({}^j\mathbf{D}_{i+1} | {}^j\mathbf{X}^{(k)}_{i+1})$ of each latent state is computed and used to reweight

the particles, with the total weight for each particle given by the product of the partial weights of each cell.

The posterior probability of the model parameters conditional on this set of simulated trajectories (assuming conditionally independent gamma priors for each parameter) is shifted similarly to in (5.7), where the α parameter increases by the summed number of reaction firings and β by the summed integrals of the (rescaled) propensity functions, over *all* trajectories until the current timepoint $i\Delta t$. I define the set $\mathcal{A}_i$ to be the set of indices of all cells observed at any time until $i\Delta t$:

$$\mathcal{A}_i = \left\{ j | {}^j\mathbf{T} \cap \{i'\Delta t\}_{i'=0}^{i} \neq \emptyset \right\} \tag{5.12}$$

Let ${}^jr_p(i\Delta t)$ be the number of firings of reaction p in cell j up until time $i\Delta t$, and ${}^jG_p(i\Delta t) = \frac{1}{k_p}\int_{{}^jt_1}^{\min({}^jt_{N_j}, i\Delta t)} a_p({}^j\mathbf{X}(s))ds$ be the integral of the propensity function divided by its respective reaction constant for reaction p and cell j up until time $i\Delta t$, for a particular realization of the stochastic process for cell j. With these definitions, the posterior joint density of model parameters $\boldsymbol{\theta}$ is given by:

$$P(\boldsymbol{\theta}|\mathcal{B}_i) = \prod_{p=1}^{d} \mathrm{Ga}(\alpha_p + \sum_{a\in\mathcal{A}_i} {}^ar_p(i\Delta t), \beta_p + \sum_{a\in\mathcal{A}_i} {}^aG_p(i\Delta t)) \tag{5.13}$$

where the set $\mathcal{B}_i = \left\{ {}^a\mathbf{X}_{[{}^at_1, \min({}^at_{N_a}, i\Delta t)]} \right\}_{a\in\mathcal{A}_i}$ gives the set of realizations of the stochastic process for all cells observed at or before time $i\Delta t$. Eq. (5.13) provides the means to generate samples $\boldsymbol{\theta}^{(k)}$ from the probability density of model parameters conditional on a particular sampled complete genealogy $\mathcal{B}_i$. The parameter samples $\boldsymbol{\theta}^{(k)}$ for each particle k are obtained by substituting the sampled trajectories for that particle into all expressions, i.e. ${}^a\mathbf{X}$ becomes ${}^a\mathbf{X}^{(k)}$, ${}^ar_p$ becomes ${}^ar_p^{(k)}$, and ${}^aG_p$ becomes ${}^aG_p^{(k)}$. Since only the summary statistics are necessary to compute the posterior of the parameters, the full trajectories are not saved, leading to a significant reduction in storage requirements. Finally, after iterating through all timepoints, the particles are resampled according to their weights yielding a set of K latent trajectories (if stored) and parameter sets. Thus the tree-based inference algorithm extends the single-cell-based inference algorithm (Algorithm 1) by establishing continuity between mother and daughter cells and initializing new latent trajectories for daughter cells according to the division process (5.11). The method constitutes an exact Bayesian inference algorithm utilizing a Gibbs sampler to generate (nearly) exact samples from the underlying stochastic process (via forward simulation using SSA or τ-leaping), and exact samples from the conditional distribution of the model parameters assuming independent gamma priors.

Algorithm 2: Tree-based recursive particle filter

Data: A set of observations $\mathcal{D} = \left\{({}^{j}\mathbf{D}_0, \ldots, {}^{j}\mathbf{D}_{N_j})\right\}_{j=1}^{N}$ of N_c cells observed at timepoints $\mathcal{T} = \left\{{}^{1}\mathbf{T}, \ldots, {}^{N}\mathbf{T}\right\} = \left\{({}^{j}t_1, \ldots, {}^{j}t_{N_j})\right\}_{j=1}^{N}$; parameter prior $\pi : \mathbb{R}^d \to \mathbb{R}_{0,+}$; observation function $g(\mathbf{D}|\mathbf{X}, \boldsymbol{\eta}) = P(\mathbf{D}|\mathbf{X})$; number of particles K; measurement time interval Δt

Result: A set of particles $\left\{\left(\left\{{}^{1}\mathbf{X}^{(k)}, \ldots, {}^{N}\mathbf{X}^{(k)}\right\}, \boldsymbol{\theta}^{(k)}\right)\right\}_{k=1}^{K}$ sampled from the posterior density $P(\left\{{}^{1}\mathbf{X}, \ldots, {}^{N}\mathbf{X}\right\}, \boldsymbol{\theta}|\mathcal{D}, \mathcal{T})$

initialization;
for *k=1...K* **do**
    Sample parameter values from the prior: $\boldsymbol{\theta}^{(k)} \sim \pi(\boldsymbol{\theta})$;
    Sample initial state conditional on first observed data point: ${}^{1}\mathbf{X}_0^{(k)} \sim g^{-1}(\cdot|{}^{1}\mathbf{D}_0, \boldsymbol{\eta})$;
    Initialize particle weight to ${}^{1}w_0^{(k)} := 1/K$
Generate a set of particle indices $\epsilon^{(k)} \in \{1, \ldots, K\}, k = 1, \ldots, K$ such that $P(\epsilon^{(k)} = a) = w_0^{(a)} / \sum_{\ell=1}^{K} w_0^{(\ell)}$;
compute maximum of all timepoints: $t_{\max} = \max(\mathcal{T})$;
loop over all observed timepoints;
for $i = 0 : \lceil t_{\max}/\Delta t \rceil$ **do**
    determine cells alive at this timepoint;
    $\boldsymbol{\sigma} = \left\{j | i\Delta t \in {}^{j}\mathbf{T}\right\}$;
    loop over particles ;
    for $k = 1 \ldots K$ **do**
        loop over cells at current timepoint ;
        for $j \in \boldsymbol{\sigma}$ **do**
            Get index of current timepoint for cell j;
            $\ell = \text{find}({}^{j}t_\ell = i\Delta t)$;
            Compute the partial weight of particle k for the j^{th} cell: ${}^{j}w_i^{(k)} = P({}^{j}\mathbf{D}_\ell | {}^{j}\mathbf{X}_i^{(k)}) = g({}^{j}\mathbf{D}_\ell | {}^{j}\mathbf{X}_i^{(k)}, \boldsymbol{\eta})$;
            Generate a sample trajectory ${}^{j}\mathbf{X}_{[i\Delta t,(i+1)\Delta t]}^{(k)} \sim P(\cdot | {}^{j}\mathbf{X}_i^{(\epsilon^{(k)})}, \boldsymbol{\theta}^{(\epsilon^{(k)})})$;
            if $(i+1)\Delta t \notin {}^{j}\mathbf{T}$ **then**
                Initialize daughter cells;
                $\left({}^{2j}\mathbf{X}_{i+1}^{(k)}, {}^{2j+1}\mathbf{X}_{i+1}^{(k)}\right) \sim P(\cdot, \cdot | {}^{j}\mathbf{X}_{i+1}^{(k)})$;
            else
                Concatenate to previously sampled trajectory: ${}^{j}\mathbf{X}_{[t_0,(i+1)\Delta t]}^{(k)} := [{}^{j}\mathbf{X}_{[0,i\Delta t]}^{(\epsilon^{(k)})}, {}^{j}\mathbf{X}_{[i\Delta t,(i+1)\Delta t]}^{(k)}]$;
        Compute the total weights for particle k: $w_i^{(k)} = \prod_{j \in \boldsymbol{\sigma}} {}^{j}w_i^{(k)}$;
        Generate a set of particle indices $\epsilon^{(k)} \in \{1, \ldots, K\}, k = 1, \ldots, K$ such that $P(\epsilon^{(k)} = a) = w_{i+1}^{(a)} / \sum_{\ell=1}^{K} w_{i+1}^{(\ell)}$;
        Generate a new set of parameters $\boldsymbol{\theta}^{(k)}$ from the conditional density: $\boldsymbol{\theta}^{(k)} \sim P(\boldsymbol{\theta} | \mathbf{X}_{[0,i\Delta t]}^{(k)})$;
Construct a sample of K particles from the posterior: $\left\{\left(\left\{{}^{j}\mathbf{X}^{(\epsilon^{(k)})}\right\}_{j=1}^{N_c}, \boldsymbol{\theta}^{(\epsilon^{(k)})}\right)\right\}_{k=1}^{K}$

Comparison to single-cell-based inference

I next tested the tree-based inference method (Algorithm 2) by performing inference on the three autoregulatory models used for testing the single-cell-based inference, namely No Feedback, and Negative or Positive Feedback, see Figure 5.2. In each case, I used the correct model to perform inference, with $K = 7 \times 10^5$ particles. I performed three replicates of each dataset to test the consistency of the inference algorithm. I find that the tree-based algorithm provides fairly consistent estimates of the model parameters for each dataset. Furthermore, the tree-based inference shows substantial improvement compared to the single-cell based inference for some model parameters, e.g. $k_{\text{on}}, k_{\text{off}}, d_m, k_p$ and d_p for the No Feedback model (Figure 5.6A); k_m and k_p for the Negative Feedback model (Figure 5.6B); and k_{on}, k_m and d_p for the Positive Feedback model (Figure 5.6C), compare with Figures 5.3, 5.4 and 5.5.

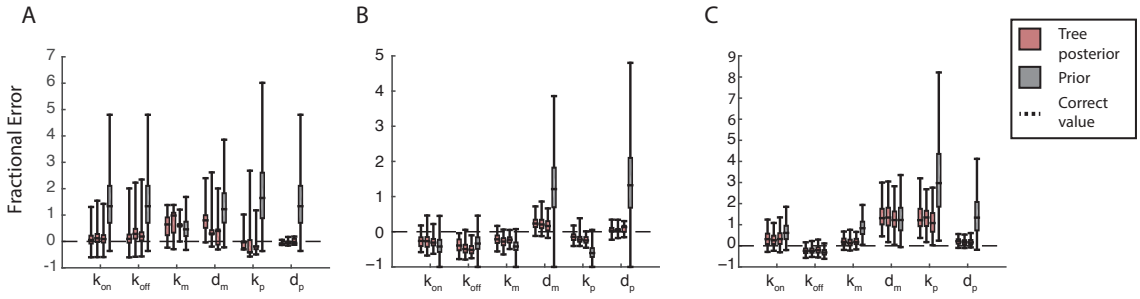

Figure 5.6: Fractional error (relative to the true parameter values) of the tree-based inference for the A. No Feedback, B. Negative Feedback and C. Positive Feedback models. In each case inference was performed using the correct model. Boxes indicate the interquartile range and whisker extend to include 95% of the sampled parameters. Gray boxes indicate the quantiles of the fractional error for the priors.

Bayes Factors

Next, I compared the marginal likelihoods of each of the three models against each of the three simulated datasets to evaluate the performance of the tree-based inference algorithm for model selection, see Table 5.7. I found that, unlike the single cell-based inference algorithm, the tree-based algorithm always selects the correct model, with Bayes Factors consistently providing very strong evidence for the correct model.

In summary, the tree-based algorithm (Algorithm 2) provides an exact Bayesian inference scheme which exploits the latent division process to enhance parameter inference and model selection for partially and discretely observed colonies of proliferating cells. The algorithm provides better estimates of parameters than is obtainable by inference using single cells, and results in stronger and more robust evidence for the correct model when doing model selection using simulated data, even for $K = 7 \times 10^5$.

	Fitted Model		
True Model	No Feedback	Negative Feedback	Positive Feedback
No Feedback	0	-43.6573 (6.4096)	-41.6947 (2.9166)
Negative Feedback	-19.6024 (0.456)	0	-56.3625 (1.8528)
Positive Feedback	-1.5769 (0.1997)	-18.9734 (0.8944)	0

Table 5.7: Log Bayes Factors for the tree-based inference, relative to the best model. Mean (standard deviation) from three replicates.

5.5 Conclusions and outlook

In this chapter I introduced an (asymptotically) exact Bayesian framework for inferring model parameters on partially and discretely observed tree-structured datasets, arising e.g. in the context of time-lapse fluorescence microscopy. I performed inference for three synthetic datasets differing in their mode of transcriptional control, showing either no feedback, or positive or negative feedback. The effect of the feedback was to increase the propensity of the activation and inactivation rates of the DNA for the Positive and Negative Feedback models, respectively. For each model I performed inference using all three possible models and showed that the tree-based inference algorithm correctly identifies the generative model in each case using Bayes Factors for model comparison. Moreover, the inference algorithm is superior in terms of inference of model parameters compared to inference based on a single-cell bootstrap particle filter, which does not incorporate information about the cellular genealogy.

Until now, I have implemented only the three simple models described. However, one can easily extend the inference procedure to incorporate a variety of additional models, such as models with post-transcriptional control (i.e. protein translation or degradation rates that depend on the protein quantity), switching models with a fixed probability of transitions between modes with different dynamical behaviors [31], models with both transcriptional and translation control, time-dependent rate constants (e.g. capturing cell-cycle-related increase in transcription rates [232]), etc. Thus, the tree-based particle filtering algorithm discussed presents a flexible framework for exact Bayesian inference (optionally with approximate forward simulation), useful for partially-observed, tree-structured cellular lineages. This method will thus become more relevant as tracked and quantified fluorescence data for cellular genealogies become more readily available.

The bootstrap particle filter is well established in the context of inference for chemical reaction networks with discretely-observed time series. However, it has not previously been extended to cellular genealogies. The extension to genealogies improves the inference procedure by forcing particles to have high likelihood for all of the cells on the tree, thereby constraining the inference procedure more than if the inference were to be performed on the cells independently of one another. Moreover, by including the tree structure, the inference procedure generates initial conditions from the stochastic process described by the assumed reaction network and binomial division process, a significant advantage over sampling independent initial conditions for the DNA and mRNA states of cells which would not accurately reflect their biological inter-dependencies.

However, some challenges remain for utilizing the particle filtering approach presented

in this chapter. In particular, the initial conditions are generally unknown, but may nonetheless have an impact on the performance of the filtering algorithm. In analyses presented in this chapter, I assumed equal probability for DNA activation and inactivation, and a random mRNA copy number less than 50 for the initial founder cell of each genealogy. The choice of initial condition is arbitrary, and a poor choice might lead to a high rejection rate of the trajectories. However, worse initial conditions will give rise to worse fits for the observed protein trajectories, and thus will eventually be filtered out by the inference algorithm. Nonetheless, a better inference result is expected for a better choice of initial conditions. In a related problem, the exact time of division is generally not known, since it may often fall between measurement times. Thus, some additional error is introduced by assuming that the division takes place exactly at the last measurement of the mother cell. However, if the observations are sufficiently frequent, this approximation error should be small.

Another challenge is presented by the choice of the scaling factor for converting measured fluorescence intensities into actual protein numbers. If possible, the calibration factors can be obtained via additional experiments, e.g. Western blot analysis to estimate the actual number of proteins present for cells with different fluorescence intensities. The scaling factor, however, greatly influences the values of the estimated protein translation and/or degradation rates. If feedback is involved, then one would expect the estimated activation/inactivation constants to be affected as well since the total rates depend on the absolute number of proteins available. It is not directly possible to infer the scaling factor along with the remaining chemical kinetic rate constants, since the filtering procedure requires the data to be unchanging from one iteration to the next. However, an alternative implementation of the tree-based algorithm in which the particle filter for the latent (tree-based) trajectories are embedded within a regular Metropolis Hastings MCMC sampler (see Section 2.2.6), may provide a framework for incorporating the unknown scaling factor. The implementation of such a particle-filtering-within-MCMC algorithm is left for future investigation.

Choice of the prior distribution of the model parameters is also worth detailed consideration. Of course, one should attempt to incorporate existing biological knowledge into the distributions, by e.g. choosing the mode of the distribution to correspond to a point estimate of the parameters (such as degradation constants). If more knowledge exists about the full distribution, variance, etc. of the rate constants, these can be included by tuning the shape of the prior distributions via the two parameters α and β. However, there is no *a priori* justification for the choice of a gamma distribution other than that this simplifies the later inference procedure. Moreover, gamma distributions are flexible enough to emulate a more correct distribution if one is known. For the tests I performed on the synthetic datasets, I attempted to choose priors that contained the correct model parameters, and were not centered on the correct parameters, so as to test convergence of the algorithm. Except for the DNA activation and inactivation rates, the parameters were chosen the same across models. Thus, for a given latent DNA trajectory, the models behave int the same way for mRNA and protein dynamics. Although this is probably sufficient for the testing, it remains nonetheless essential to carefully choose priors as best reflects biological knowledge, so as not to bias the inference procedure.

The inference algorithm assumes binomial distribution of protein and mRNA to the

two daughter cells. The assumption of nearly equal distribution of proteins is in good agreement with the observed dataset, whereas the mRNA distribution is merely an assumption. The assumption that the DNA state is persistent from mother to daughter is stronger, and is motivated by the observation in other NanogVENUS datasets that many progeny will begin to show increased protein production rate after a few generations of quiescence, suggesting a switch-like behavior and persistence in cellular offspring. This assumption is not essential to the working of the algorithm, since it is only used to sample the initial state of the daughter cells conditional on the mother cell, and can thus be easily relaxed or modified.

In the case of time series obtained from fluorescence microscopy, additional uncertainty is introduced by the unknown protein measurement error. In each of the preceding applications, I have assumed Gaussian measurement error with a fixed variance. However, such an assumption is not essential to the algorithm, as it is only necessary for computing the weights of each particle at each iteration of the inference procedure (see Algorithm 2). In principle any other (non-Gaussian) observation function could be used as well. Additionally, although the magnitude of the measurement error can in principle be inferred from the data, such an extension has not been considered in the present work. In particular, it may be possible to infer the noise magnitude by including it as a hyperparameter in a particle-filter-within-MCMC context, as previously mentioned. Critically, assuming too small a measurement error can cause the particle filter to reject many more particles, and perhaps become degenerate as only a few particles are retained. This problem is partly alleviated by having a very large number of particles for inference. Conversely, too large a measurement error can cause too many particles to be accepted and thus the particle filter will not remove particles that are incompatible with the data, leading to lack of convergence to the correct model parameters. Thus, the measurement error should be carefully chosen to be as close to the real value as possible.

The inference procedure is obviously limited by the quality of the data. For example, very noisy data contain less information about the model parameters and latent trajectories, and thus limit the ability to infer parameters and model structure. Also, the performance of the algorithm depends on the values of the true model parameters underlying the data—if the parameters are such that little information is contained in the observed trajectory, e.g. if the mRNA is so stable that it effectively averages the DNA activity state, then the inference procedure will not be able to deduce the latent DNA trajectory. It is not clear when exactly this is to be expected, and thus this warrants a more thorough investigation. However, it is hoped that the inclusion of longer, broader (in the sense of more cells observed) genealogies will somewhat alleviate this problem.

The number of particles used in the inference procedure is also of importance. If too few particles are used, there is the possibility of degeneracy whereby only a few particles are sampled repeatedly, leading to a biased estimate of the posterior distribution. In principle, the more particles the better, but this of course incurs additional computational overhead. In practice, $10^5 - 10^6$ particles seemed to suffice for the present analyses. The single-cell-based inferences for the synthetic data were carried out with 10^5 particles, and the tree-based inferences with 7×10^5; since there were 7 cells in each synthetic dataset the total number of particles is the same in the single-cell and tree-based cases. However, as more trajectories are simulated in parallel, i.e. as the tree branches, the

overall likelihoods and thus the weights of the particles decreases rapidly, possibly leading a higher filtering rate of the particles. Hence, it may become necessary to greatly increase the number of particles to ensure adequate sampling of the posterior. For the current analyses, computation was greatly accelerated by utilizing a multicore architecture for the forward simulations. Individual model and dataset combinations are independent and were thus run in parallel on different compute nodes of a cluster. Additional acceleration may be possible by utilizing a more efficient implementation of the forward simulation procedure, or by utilizing GPUs instead of CPUs, which may lead to a substantial time savings for trivially parallel computations.

Various extensions have been proposed to the original particle filtering algorithm of Gordon *et al.* [226]. For example, it may be possible to improve sampling efficiency by increasing the number of trajectories with high likelihoods at each iteration of the inference algorithm. This might be obtained e.g. via the auxiliary variable method of Pitt and Shephard [233] which reweights particles according to some approximation to the predicted likelihood of each particle upon the next observation. By reducing the number of particles discarded during the resampling step, the "variability" of the samples is increased. Such an approach might reduce the observed degeneracy of samples, i.e. the number of sampled latent trajectories (and thus components in the resulting mixture model) can become small if most simulated trajectories are rejected. Hence the auxiliary variable approach could reduce the number of particles needed to obtain an adequate estimate of the posterior distributions, thus increasing computational efficiency. The implementation of such a strategy is left for future work.

Lastly, although I have focused in particular on autoregulatory models for a single gene, the method is general enough to be applied for any stochastic gene regulatory model with an arbitrary number of observed species. The likelihood function can easily be modified to incorporate multiple observed species simultaneously for experiments involving more than one fluorophore. Thus, the method provides a new tool for performing inference in tree-structured timeseries data, and may serve as the starting point for the development of more advanced tree-based inference methods.

Chapter 6

Analysis of NanogVENUS cellular lineages

6.1 Introduction

In this chapter, I present the result of a set of investigations into the behavior of populations of embryonic stem cells (ESCs), at the single-cell level. As discussed in Chapter 1, mESCs are known to exhibit widely heterogeneous expression levels for some pluripotency factors, including Nanog. However, the function and regulation of the core pluripotency network of mESCs is less well understood. In particular, the mechanism underlying the emergence of Nanog heterogeneity is still poorly characterized. This is partly due to the lack of time-resolved single-cell expression data of adequate depth and breadth for Nanog's protein product. Here, we examine data generated by the Institute of Stem Cell Research at the Helmholtz Zentrum München, and provide quantitative analysis of single-cell protein expression time-courses and cellular genealogies. In the following sections, we investigate possible oscillations, Nanog intensity transitions among progeny, fate determination of sister cells, and identify a novel Nanog subpopulation based on Nanog expression and the structure of correlation networks for core pluripotency factors, using the multiresolution correlation method presented in Chapter 3. Finally, I apply the inference algorithm developed in Chapter 5 to this dataset in order to infer model parameters and the most likely autoregulatory motif for the observed data.

The results included in this chapter are partially based on the publication (for which I am second author): Filipzyck, A., Marr, C., Hastreiter, S., **Feigelman, J.**, Schwarzfischer, M., Hoppe, P. S., et al. (2015). Network plasticity of pluripotency transcription factors in embryonic stem cells. Nature Cell Biology. http://doi.org/10.1038/ncb3237

The analyses and figures contained in the sections "Oscillations", "Onsets", and "Memory" of section "Characterization of Nanog dynamics", and the sections "Identification of subpopulations" and "Stochastic autoregulatory models for NanogVENUS dynamics" are entirely my own work. The analysis in Section "Transitions" is joint work with my colleague Dr. Carsten Marr, who also produced the Figures 8-12. This Section and the figures listed are included for completeness and to closely parallel the exposition in the aforementioned publication.

6.2 Experimental setup

All analysis of Nanog protein dynamics is based on data generated by Dr. Adam Filipczyk and Simon Hastreiter of Prof. Timm Schroeder's lab at the Institute of Stem Cell Research (ISF), Helmholtz Zentrum München, and now of the Cell Systems Dynamics group at BSSE, ETH Zürich, from 2011-2014.

6.2.1 Generation of mESC fluorescent fusion protein line

Nanog protein quantity was characterized using a transgenic R1-mESC line, for which the yellow fluorescent reporter molecule VENUS was integrated into the mESC genome under the control of the endogenous Nanog promoter. The resultant C-terminally fused protein complex, dubbed NanogVENUS, was shown to be a faithful reporter of Nanog levels within the cell, as the fluorescent molecule VENUS is co-expressed with Nanog. Furthermore, the half-life of the construct is not significantly altered with respect to the wild type protein. Generation of the fusion protein and characterization of its biological activity has been described elsewhere [154, 234].

NanogVENUS mESCs were imaged using brightfield and fluorescence time-lapse microscopy at half hour intervals under a variety of conditions and durations, as described below. The aim of the experiments was to capture the dynamic behavior of individual stem cells isolated from different compartments of the fluorescence intensity distribution, possessing either low or high expression of NanogVENUS protein. The core assumption of the experimental setup is that the fluorescence intensity is proportional to the number of Nanog molecules, and thus that fluorescence microscopy provides an accurate quantification of protein levels in individual cells. In order to quantify protein levels, several steps are required, all of which are described in detail in the manuscript by Schwarzfischer *et al.* [62].

6.2.2 NanogVENUS quantification

In the first step, cells must be tracked. Tracking entails maintaining an unambiguous labeling of each individual cell in the population, from the time of birth of the cell until the time of death or division. For quantitative analysis, it is essential that the identity of the cell is known at all times. Thus cells were tracked manually using the software Timm's Tracking Tool (TTT), which provides a graphical user interface for labeling cells from time-lapse microscopy movies and the means to annotate events such as death and division. Tracking was generally performed on the brightfield channel of the microscopy setup, due to the relatively low phototoxicity that it induces; if tracking was not possible via brightfield, fluorescence channels were sometimes used instead. Using TTT, cellular genealogies up to 7 generations were established for clonal colonies of mESCs.

After tracking, individual cells must be segmented. By segmenting the cells, image regions are identified which ideally capture the entirety of the cell (or of its nuclear region in some cases), and none of the background or adjacent cells. Segmentation is possible via a variety of numerical methods, most simply via thresholding algorithms for separating foreground from background based on signal intensity. Segmentation was performed using the tool QTFy (Schwarzfischer *et al.* [62]), which provides a graphical interface for

the semi-automatic identification and quantification of tree-structured cellular genealogies derived from time-lapse microscopy. In most cases, the NanogVENUS mESCs were virally transfected with a fluorescent protein (CHERRY), fused to a nuclear membrane marker, Nucmem. The product of the construct, mCHERRYNucmem, localizes to the nuclear membrane, and thus provides a fluorescence signal of a different wavelength than the VENUS protein used for quantifying Nanog levels. The mCHERRYNucmem signal hence provides a reliable means to detect and segment the nuclei of individual ES cells, even when no Nanog is expressed, as is sometimes the case. In some instances, mCHERRYNucmem was not stably transfected, in which case segmentation was instead performed on the brightfield images as best could be achieved without a nuclear marker.

In the last step, the fluorescence intensity must be quantified for the tracked and segmented cells, which was also performed in QTFy. In the simplest case, intensity is quantified by using the cellular (or nuclear) regions as a mask on the fluorescence intensity channel. The intensity within the masked region is summed to give an approximation to the total protein content of the cell, up to a constant conversion factor between protein intensity and actual number of molecules. Additionally, the intensity must be corrected by compensating for uneven illumination of the images due to effects arising from the microscopy setup, and for pixel-specific gain functions that could otherwise bias the estimation of cellular intensities. These corrections were performed prior to subsequent analysis of quantified intensity time-courses, and were described in [62].

Several experiments were performed throughout our investigations of Nanog dynamics, see Table 6.1. Experiments are principally divided depending on the intensity of the cells used to generate the subsequent colonies. In particular, three populations were examined: cells isolated from the lowest 2% of the NanogVENUS fluorescence distribution (low-sorted), the highest 5% of the NanogVENUS distribution (high-sorted), and unsorted cells which were obtained without any flow-cytometric cell sorting. The experiments used in subsequent analysis ranged from 1 day to about 10 days in duration, with thousands of individual cells and dozens of single colonies tracked. mESC colonies derived from single stem cells as confirmed by the high dilution used when plating the cells, and by inspection of the resultant movies. In each case, mESC colonies were grown in serum/LIF, a pluripotency-promoting culture medium [7].

ID	Description	# cells	# trees	Duration (h)	Use
110907	Unsorted, unbiased	255	4	92.7	O
110930	Unsorted, unbiased	1829	22	94.0	O
111115	High-sorted	4482	22	120.4	T
111115	Low-sorted	3550	27	173.3	T, On
110930	Low-sorted	796	16	94.0	T, On, PF
110613	Low-sorted	683	7	254.0	On
120410	Low-sorted	2758	25	96	ES
120516	Low-sorted	836	17	96	ES
130503	Low-sorted	623	15	96	ES
120509	Low-sorted	76	25	24	DS
130115	Low-sorted	215	57	24	DS
130204	Low-sorted	234	68	24	DS
130208	Low-sorted	258	201	24	DS

Table 6.1: Experiments used for analysis of NanogVENUS dynamics. Legend: **O**: oscillations, **T** transition densities, **On**: onsets, **ES**: endpoint staining analysis, **DS**: divergent sisters analysis, **PF**: particle filter inference

.

6.3 Characterization of Nanog dynamics

6.3.1 Oscillations

Nanog is known to be heterogeneously expressed in mESCS under serum/LIF conditions (see Chapter 1). However, it is unclear what regulatory features induce the observed heterogeneity. One hypothesis is that Nanog heterogeneity emerges due to regular oscillations, which would have the potential benefit of allowing mESCs to explore a wide range of expression levels and prime mESCs for differentiation from the transient Nanog low state, e.g. in response to environmental signaling. Nanog oscillations could arise due to the presence of a negative feedback loop, giving rise to periodic fluctuations in Nanog expression. Alternatively, Nanog could fluctuate between two meta-stable attractors corresponding to low and high expression. Both hypotheses have been put forward by Glauche *et al.* [29], see Section 1.4. However, until now, no evidence has been presented in the literature substantiating or refuting the oscillatory hypothesis of Nanog expression in ESCs. Indeed, evidence for oscillations by definition can only be discerned from time-resolved Nanog expression, for which there have been few studies until now, largely due to the lack of a fluorescent reporter for Nanog's protein product. For example, Abranches *et al.* [235] developed the VNP Nanog protein reporter, which expresses a destabilized Nanog construct via the insertion of a bacterial artificial chromosome, while leaving the two endogenous Nanog alleles intact. In the most recent work, Abranches *et al.* [26] utilize this system to perform single-cell time-lapse fluorescence microscopy under serum/LIF and 2i conditions. However, their data are limited to a single-cellular generation precluding the detection of oscillations over multiple generations. Furthermore, they perform no analysis for the detection of oscillations, nor do their data suggest oscillations upon inspection.

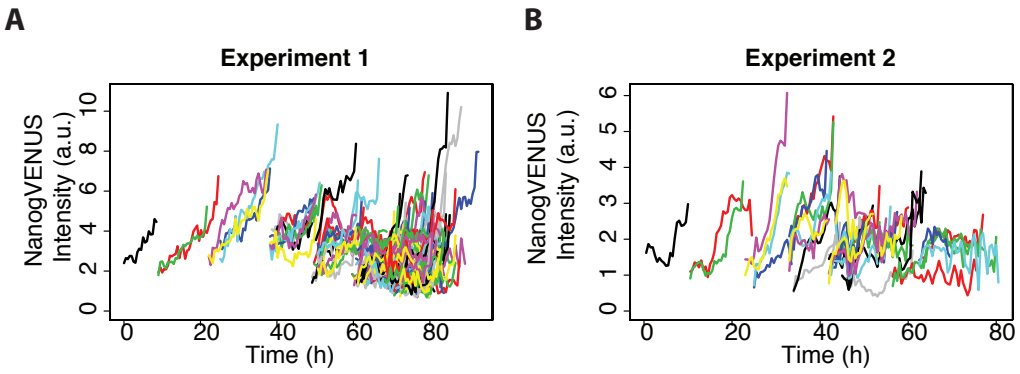

Figure 6.1: Two fully-inspected NanogVENUS ESC colonies used for investigation of NanogVENUS dynamics. single-cells are distinguished with random colors.

Besides Abranches, Miyanari *et al.* constructed a two-color (turboGFP/mCherry) Nanog protein fusion mESC line [153]. They concluded that Nanog is largely monoallelically expressed during embryonic development. However, this view has been contested, raising additional uncertainty [34, 154]. Hence, Nanog expression dynamics at the protein level has not thus far been thoroughly characterized, nor has any evidence been provided that might refute hypotheses pertaining to possible oscillations, motivating the subsequent analyses.

To investigate potential oscillations in the NanogVENUS dataset, I examined data from two datasets generated by our collaborators, that were *unbiased*, i.e. not enriched for low or high NanogVENUS expression, and tracked and quantified agnostically of cell fate in order to prevent bias in the subsequent analysis. The NanogVENUS expression time series for single-cells in these two data sets—Experiment 1 (ID 110907) and Experiment 2 (ID 110930)—are shown in Figure 6.1. For this investigation I utilized only data that had been fully inspected manually using QTFy in order to minimize the chance of error arising due to mis-segmentation.

In general, oscillations in a univariate time series may be detected using the autocorrelation of the signal, see Section 2.3.2. For example, an oscillator (a sinusoid with Gaussian noise added) with a fixed period T shows a significant positive correlation at lags of nT, $n = 1, 2, \ldots$, and a negative correlation at a lag of $T(n + 1/2)$, $n = 0, 1, \ldots$, see Figure 6.2. Thus, an easy way to detect oscillations in the NanogVENUS signal is to inspect single cells and look for periodicity in the autocorrelation function corresponding to the period of NanogVENUS oscillations. However, by examining the autocorrelation of individual cells (see Figure 6.3), it is immediately clear that although there are a wide variety of behaviors exhibited, there is little to suggest oscillations, at least, not within the lifetimes of individual cells. However, it is possible that a cellular *lineage* could exhibit oscillations over several generations.

To search for multi-generational oscillations, I compute the autocorrelation for each branch in each of the two experiments, and combine them to obtain statistics for the autocorrelation over generations. In Figure 6.4, I display the mean, 25% and 75% quantiles of the autocorrelation, for varying lags, for both experiments separately. While Figure 6.4A

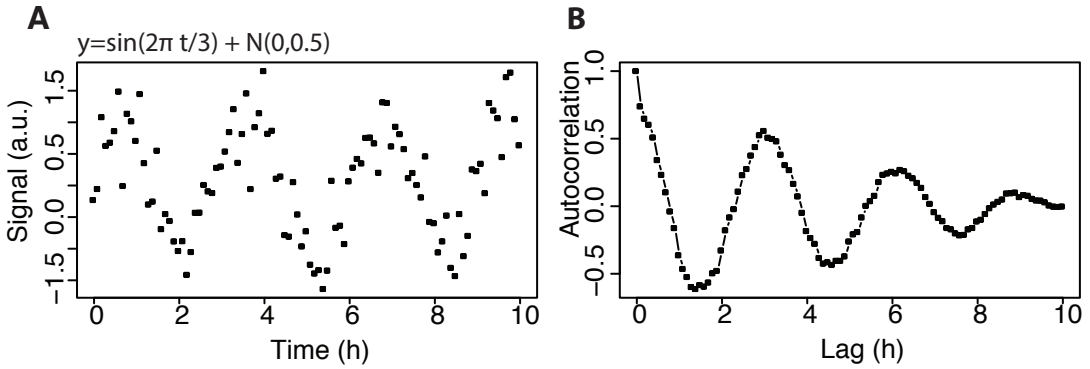

Figure 6.2: Autocorrelations can be used to detect oscillations. A. Example of a simple sinusoidal oscillator with period $T = 3$ with Gaussian error. B. The corresponding autocorrelation function shows clear peaks at multiples of the period.

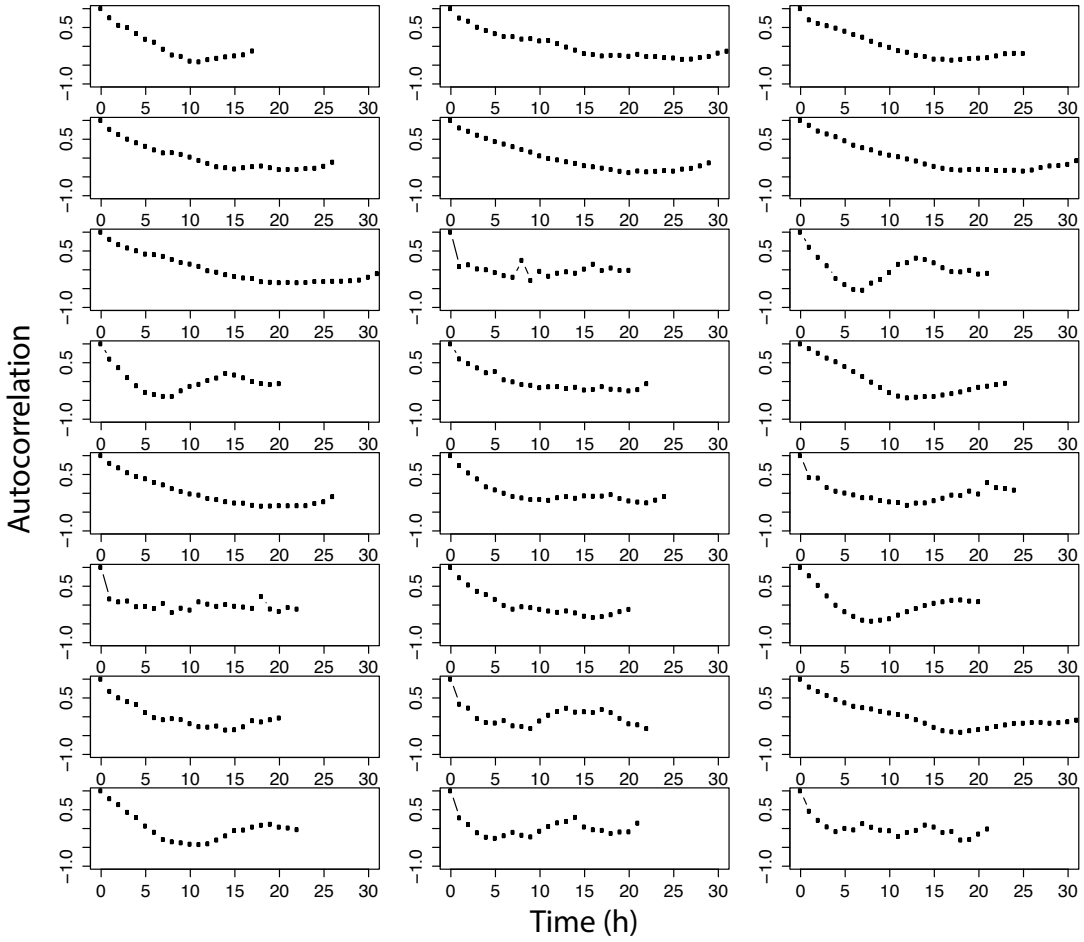

Figure 6.3: Autocorrelations computed for single-cells from Experiment 1 suggest lack of stable oscillations. Here only a subset of $N = 24$ randomly chosen cells are shown.

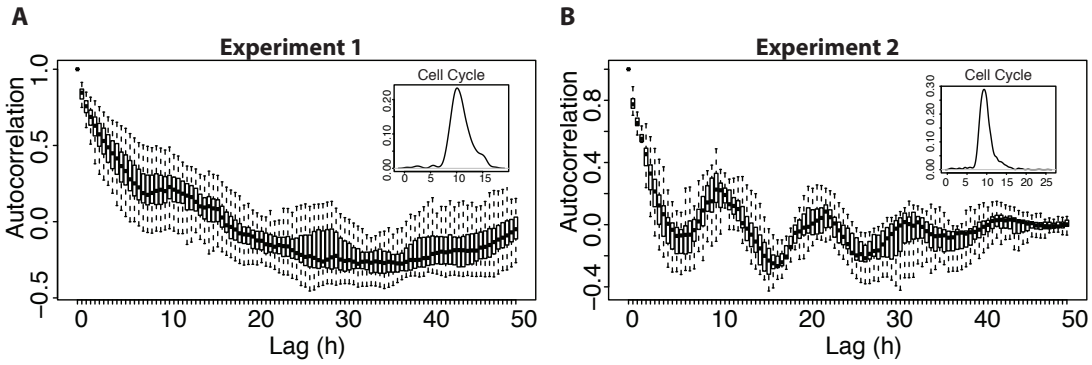

Figure 6.4: Autocorrelation functions for all branches in each of two experiments, shown as median and 25%, 75% quantiles. A. Experiment 1 shows a trend of decreasing autocorrelation with increasing lag, suggesting no oscillations. B. Experiment 2 shows seeming periodic peaks in the autocorrelation suggesting oscillations with period of about 10 hours, possibly due to the cell cycle-related increase in NanogVENUS intensity which peaks sharply at 10h. Cell cycle length distribution shown in inset for both experiments.

shows no evidence for oscillations, Figure 6.4B would seem to indicate that Experiment 2 shows oscillations with period of about 10 hours. However, the cells naturally oscillate with a period of about 10 hours due to the cell cycle (inset), hence it is very likely that these apparent oscillations arise due to this effect. Furthermore, the cell cycle is more sharply peaked for Experiment 2 (coefficient of variation of 1.94 vs 2.14 for Experiment 1), which could explain the stronger degree of cell-cycle related effect visible in Figure 6.4B.

One approach to try to remove the cell cycle effect from the NanogVENUS signal entails normalizing the signal by the approximate nuclear volume to achieve the NanogVENUS intensity concentration at each timepoint. The reasoning behind this is that both absolute NanogVENUS copy numbers and nuclear volume are increasing in time. If the cell attempts to maintain constant concentration of Nanog, and hence constant action on Nanog's downstream targets, then Nanog protein (and hence NanogVENUS signal) should increase proportionally to nuclear volume. Moreover, it has recently been shown that mammalian cells scale transcription to maintain an approximately constant concentration in transcripts for varying cellular volumes [232]. Thus it is potentially possible to remove the cell cycle-dependent increase in NanogVENUS intensity simply by dividing by the estimated nuclear area, obtained from segmentation of the nuclear marker mCHERRYnucmem, which serves as a proxy to the nuclear volume. ES cells in culture are typically flat due to the cadherin-functionalized surface in the culture plate, suggesting that the proportionality constant between area and volume is roughly constant. However, examination of the nuclear area-normalized intensity concentrations reveals a residual cell cycle dependent increases near the time of division, see Figure 6.5. This is presumably due to the "rounding" of the ES cells prior to division. The rounding leads to an increased volume/area ratio, and conversely to an apparent increase in the intensity/area concentration.

While the cells inherently oscillate due to the cell cycle, the intensity nonetheless

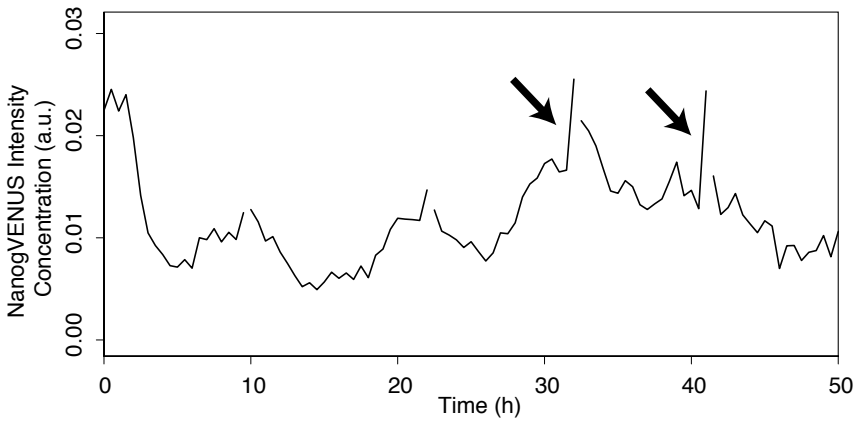

Figure 6.5: Normalization of the NanogVENUS intensities along cellular lineages fails to completely abate cell-cycle-dependent oscillations. Prominent artifacts are visible near the point of division (indicated with arrows) due to rounding morphology near the time of division, illustrate with one lineage from Experiment 2.

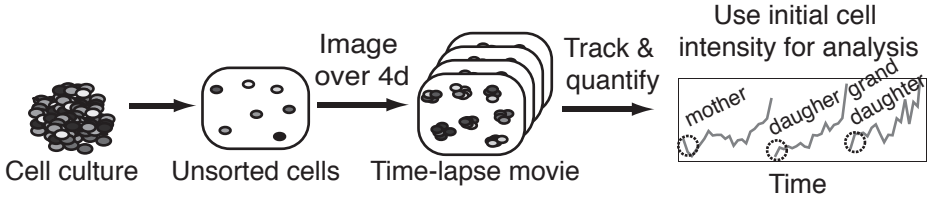

Figure 6.6: Unsorted cells are taken, plated and imaged over a period of 4 days. Cells were tracked, quantified, and the median intensities of the initial three timepoints used for analysis of oscillations.

remains fairly stable from one generation to the next in steady state. Thus, a more suitable measure for detecting multi-generational oscillations may be to examine the intensity at the time of birth of each cell along a lineage. It is thus possible to compute the autocorrelation of a set of statistically independent branches, using only the intensity at the beginning of the time series for each cell. To reduce the effect of measurement error, the median of the first three timepoints of each cell is used instead of the first timepoint alone, see Figure 6.6.

I identified a set of statistically-independent branches from Experiment 2 by filtering for branches of at least 5 generations in order to resolve potentially multi-generational oscillations, while retaining at most one branch per subtree—i.e. no cells in the retained branches were common to more than one branch; Experiment 1 contained just two such branches and was not used for this analysis. However, with this restriction the amount of available data becomes limited, since from the potentially $2^4 = 16$ branches that may derive from a single ancestor cell over 5 generations, only one is retained for the analysis to ensure statistical independence amongst branches. Thus, the filtering process resulted in 16 total suitable branches, see Figure 6.7A. From these branches it is evident that oscillations are not a dominant behavior of the cellular lineages. While some branches

show non-monotonic behavior, the magnitude of the fluctuation is typically on the order of one intensity unit or less. Moreover, none of the fluctuations are sufficient to drive the expression level below the detection limit of 0.101 units, suggesting that oscillations do not cause ES cells to enter the Nanog- compartment as previously hypothesized in [29]. Similarly, examination of the autocorrelations of each of the branches shown confirms lack of oscillation, Figure 6.7B.

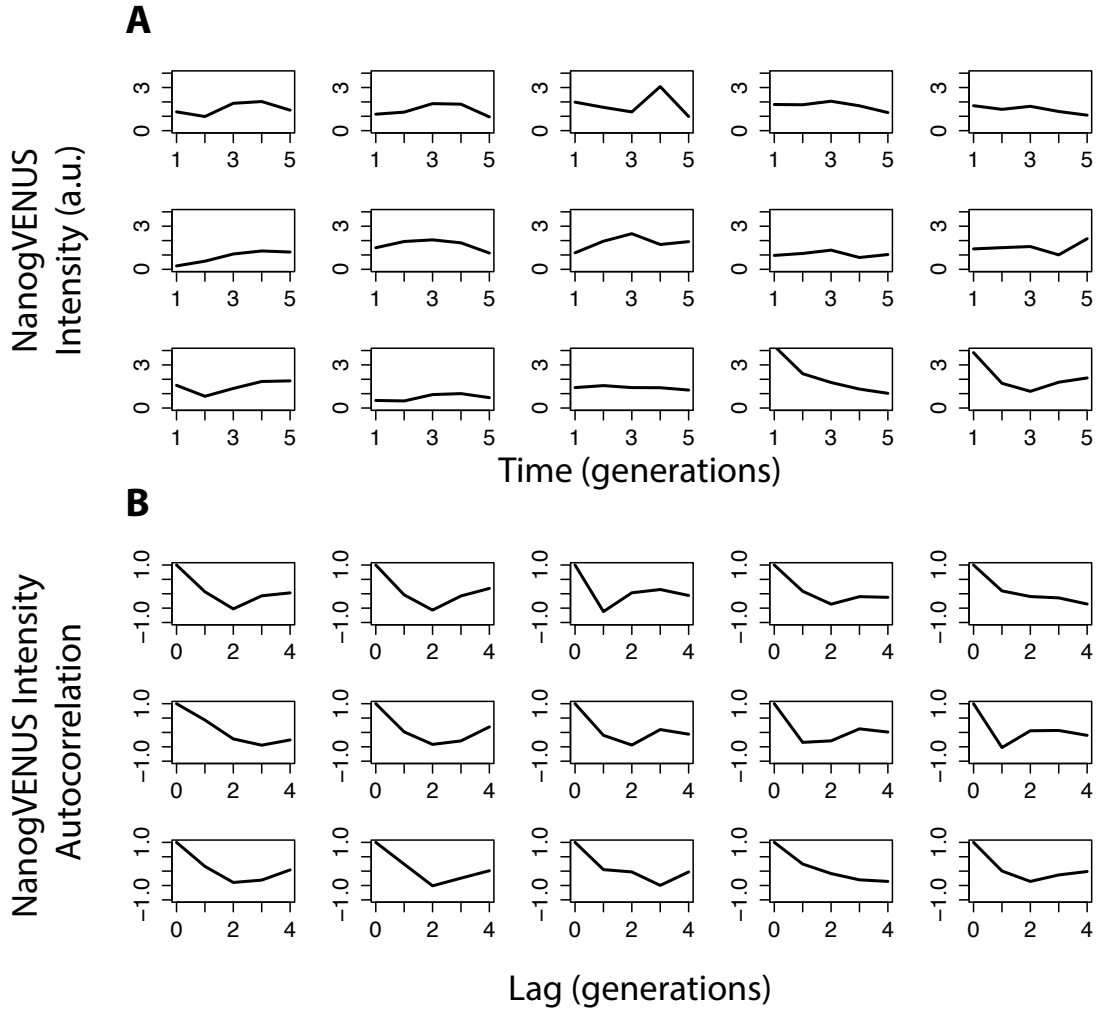

Figure 6.7: A. Statistically-independent branches generated from Experiment 2, keeping only the median NanogVENUS intensity of the first three timepoints of each generation. No obvious oscillations exist over after removing cell-cycle driven increase. B. Autocorrelations of the branches shown in (A). No clear periodic peaks exist, suggesting lack of multi-generational oscillations.

6.3.2 Transitions

Next, to help characterize the NanogVENUS intensity transitions observed within mESC colonies, I define 4 intensity compartments: the Negative compartment contains intensities that fall below the 95% quantile of the estimated background intensities, and thus represent no detectable NanogVENUS signal; the remaining three intensity compartments (Low, Mid, High) equally divide the log-transformed NanogVENUS intensity range, see Figure 6.8. Furthermore, by examining the NanogVENUS intensity only at the beginning of each observed cellular trajectory, the cell-cycle driven increase in expression is removed, leading to the "cell-cycle corrected" expression profile shown in Figure 6.9. By correcting for the cell cycle, we are able to artificially synchronize cells in a population in order to facilitate comparison of expression intensity.

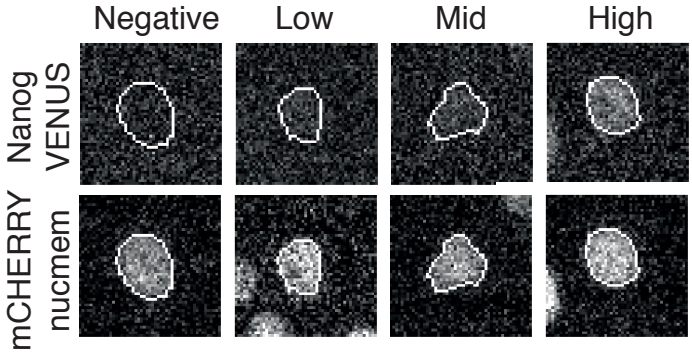

Figure 6.8: The NanogVENUS intensity expression range was divided into four compartments. Negative cells have no detectable NanogVENUS expression; nonetheless the cells are still detected using the mCHERRYnucmem nuclear marker. The Low, Mid, and High compartments correspond to the bins of equal size spanning the log-transformed NanogVENUS intensity range.

I next computed the estimated transition density from initial mother cell intensity to initial daughter cell intensity. For this I use a simple kernel density estimator (KDE) to approximate the density of observed mother-daughter intensity transitions present in Experiments 1 and 2, see Figure 6.10. We observe an apparent bistability, with high cells preferentially remaining high, and low cells preferentially remaining low, as evidenced by the increased density in the lower left, and upper right corners of the plot, respectively. This bistability is reminiscent of the bistabilities previously reported [24, 235]. Furthermore the majority of cells give rise to progeny with initial NanogVENUS expression intensities between 1/2 and 2 times the their own initial intensity (dashed diagonal lines).

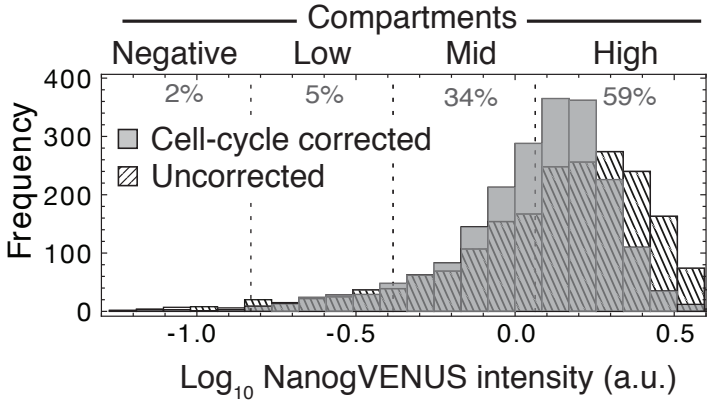

Figure 6.9: The cell-cycle corrected NanogVENUS intensity distributions show reduced expression when the cell-cycle-associated intensity increase is removed.

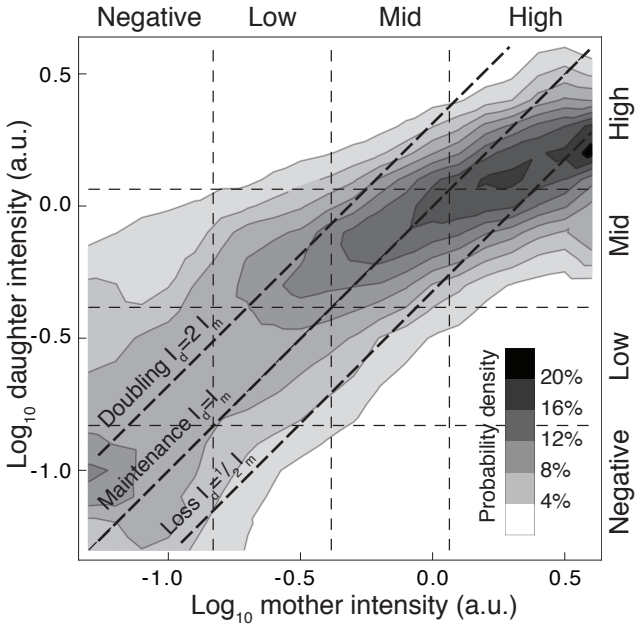

Figure 6.10: The empirical transition kernel between initial mother cell intensity, and daughter cell intensity shows increased density consistent with high cells preferentially remaining high, and low cells preferentially remaining low. Cells rarely give rise to progeny with intensity less than half or more than double the initial intensity of the mother cell (dashed diagonal lines).

Markov model

Utilizing the estimated transition kernel (Figure 6.10), it is possible to predict the (log) intensity distribution at the next generation, simply by performing the vector-matrix multiplication of the estimated density at each generation with the transition kernel density matrix. Interestingly, the predicted distributions for populations sorted from each of the defined intensity compartments show very good qualitative agreement with the actual distributions arising from the sorted cells, as determined by time-lapse fluorescence microscopy, see Figure 6.11.

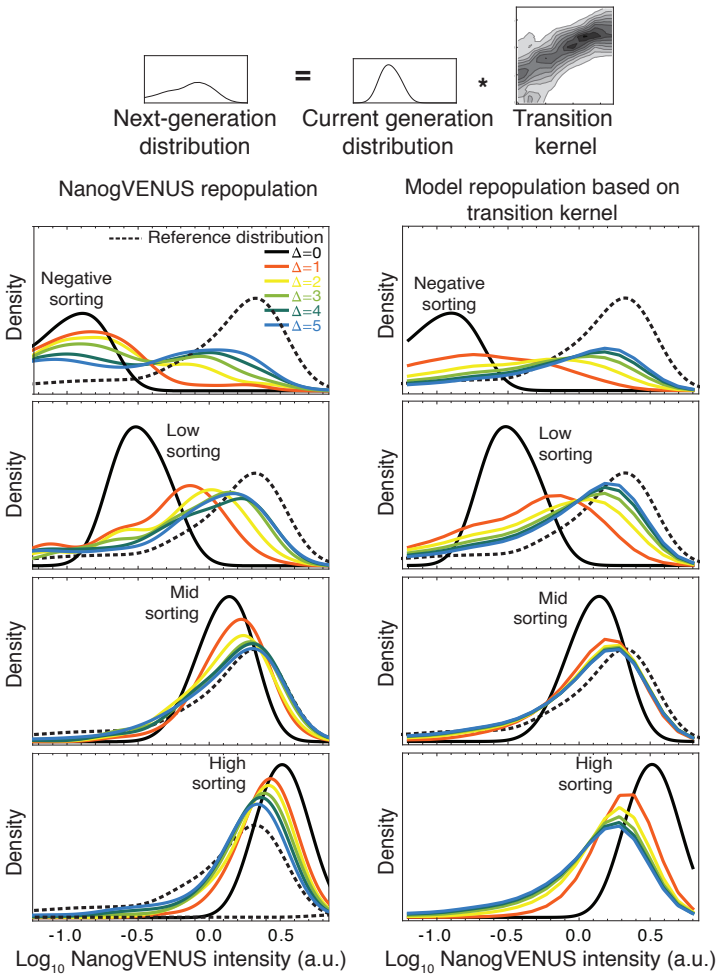

Figure 6.11: The Log_{10} NanogVENUS intensity distribution at $\Delta = 0, \ldots, 5$ generations subsequent to sorting into one of the four intensity compartments is well approximated by a simple Markov model. The intensity at the next generation is the vector-matrix product of the intensity distribution at the current generation (approximated using a KDE), and the empirical transition density (see Figure 6.10).

While at the population level the dynamics are seemingly well approximated by a simple Markov model, the dynamics of individual cells are much less predictable, as is

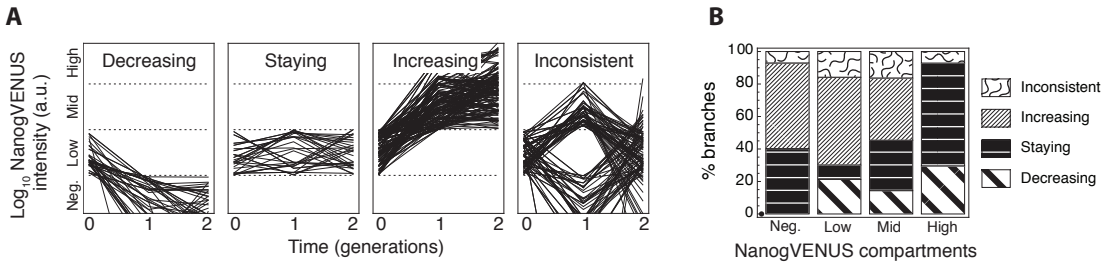

Figure 6.12: Low-sorted mESCs undergo a variety of dynamical behaviors when examined at the single-cell level. A. We characterize the behavior of individual cellular lineages over three generations as belonging to one of four categories: decreasing, staying, increasing, or inconsistent, based on the intensity compartments of the progeny cells. B. The negative and high compartments show an increased fraction of branches remaining in the same intensity category after two generations, relative to the low and mid intensity compartments, consistent with the estimated bistable transition density (Figure 6.10). Cells from the lowest compartment show a predominance of increasing intensity, while cells from higher compartments tend to decrease in intensity.

apparent when plotting the transitions for two consecutive generations of a collection of low-sorted cells, see Figure 6.12. Here, low-sorted cells may either remain low, decrease, increase, or be inconsistent, e.g. increase or decrease followed by the opposite, see Figure 6.12A. While the majority of low-sorted cells show increasing intensity in progeny, consistent with the repopulation of the distribution shown in Figure 6.11, there are nonetheless many inconsistent and decreasing cellular lineages as well; see Figure 6.12B for summary.

6.3.3 Onsets

It is evident that single ES cells are capable of a wide range of dynamic behaviors, (see Figure 6.12), thus we decided to further investigate the NanogVENUS dynamics of individual cells and branches or lineages within low-sorted colonies. We noticed in particular that colonies often continue to have low NanogVENUS expression for a small number of generations, after which most colonies begin to exhibit increased NanogVENUS intensity—presumably due to upregulation of NanogVENUS expression—in one or more cells, see e.g. Figure 6.13. This transition from low to high is critical to the mESCs' ability to restore the steady state distribution upon sorting. Thus, I analyzed the dynamical behavior of low-sorted cells undergoing this transition.

Generalized logistic function

In order to characterize the dynamics of individual "onsets", i.e. the point where NanogVENUS intensity begins to show a rapid increase, I identified a set of 42 onset-containing cellular lineages derived from low-sorted mESCs via manual inspection (see Table 6.1). To establish continuity between cells in a cellular lineage, and to minimize the effect of the cell-cycle on NanogVENUS intensity, I divide the intensity of each cell by the estimated nuclear area (obtained via the mCHERRYnucmem signal), to obtain the average per-

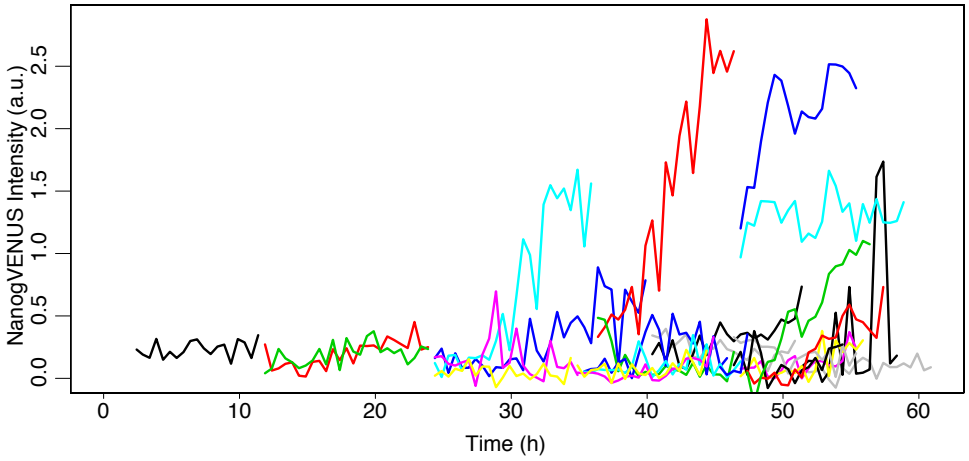

Figure 6.13: Example low-sorted NanogVENUS colony emerging from a single progenitor cell. Low-sorted NanogVENUS trees typically show onset of NanogVENUS production within a few generations following sorting. Individual cells are distinguished using random colors.

pixel NanogVENUS intensity, or intensity concentration time series. Note that averaged NanogVENUS time series sometimes still show artifacts near division, as previously mentioned (see Figure 6.5). I then fit each of these time series by a simple sigmoid function, given by the generalized logistic function:

$$Y(t) = A + \frac{K - A}{(1 + Qe^{-B(t-t_0)})^{1/\nu}}. \tag{6.1}$$

The parameters can be interpreted as follows: K gives the upper asymptote, A the lower asymptote, B the growth rate, ν pertains to the shape of the sigmoid, t_0 is a time parameter, for which $Y(t_0) = A + \frac{K-A}{2^{1/\nu}}$, and Q is related to the value $Y(0)$. Thus the generalized logistic function is a flexible sigmoidal function, from which one can directly ascertain relevant biological features such as maximal and minimal expression levels, rate of growth, time of onset, etc.

The generalized logistic function was fit to each of the 42 identified onsets, to obtain the fits shown in Figure 6.14. Fitting was accomplished by least-squared error minimization of the model for each fit individually, using 100 restarts initialized using Latin hypercube sampling with parameters bounded by the constraints given in Table 6.2. Constraints were chosen to be very flexible, i.e. wide but arbitrary upper and lower bounds; the estimated parameters of the resulting fits were inspected to ensure that the fits were not biased due to the constraints.

For each fit, one obtains immediately the estimated minimal averaged NanogVENUS intensity (A), and the maximal averaged NanogVENUS intensity (K). By solving the second derivative of (6.1) for its root, i.e. computing $Y''(t_{\max}) = 0$, one obtains the time $t_{\max}$ at which the slope $Y'(t)$ is maximal. This slope, denoted $Y'_{\max}$, represents the maximal growth rate of the averaged NanogVENUS intensity. Lastly, the parameter t_0 reflects the relative shape of the onset sigmoid, with small t_0 being left-shifted (i.e. early), and large

	A	K	B	ν	Q	t_0
Min.	0	0	0	0	0	0
Max.	2	10	100	100	100	100

Table 6.2: Parameter constraints for fitting the onset to the generalized logistic function (6.1).

t_0 being right-shifted (late). Using the fitted curves, we obtain the statistics summarized in Table 6.3. Clearly the identified onsets are very heterogeneous in all parameters except for the lower asymptote, which is similar across all onsets since the lineages begin with very low or non-detectable NanogVENUS expression.

	Lower Asymptote (A)	Upper Asymptote (K)	Max Growth Rate ($Y'_{\max}$)	Onset Time (t_0)
Mean	0.0012	0.1412	47.0086	0.0188
Std	0.0008	0.5209	23.7081	0.1006

Table 6.3: Fitted averaged NanogVENUS intensity onsets in low-sorted mESCs reveal substantial heterogeneity in estimated maximum intensity concentration, rate of increase, and time of onset within the fitted onsets.

ODE model

As an alternative to the generalized logistic model, I also fit the onset curves using a simple differential equation model for the mRNA and protein concentrations, assuming that the DNA is initially in an inactive configuration and becomes active at an unknown switch time, at which point transcription proceeds with a greater rate, see Figure 6.15.

The ODE model treats the mRNA and protein numbers as deterministic variables with unknown switching time t_0:

$$\begin{aligned} \frac{dR}{dt} &= \begin{cases} k_{r_0} - \gamma_r R, & t < t_0 \\ k_{r_1} - \gamma_r R, & t \geq t_0 \end{cases} \\ \frac{dP}{dt} &= k_p R - \gamma_p P \end{aligned} \tag{6.2}$$

which corresponds to constant production of mRNA with rate constant that switches at t_0, production of protein proportional to mRNA, and linear degradation of mRNA and protein with rate constants γ_r and γ_p, respectively. The model does not explicitly incorporate cell cycle, however.

The system (6.2) was solved analytically and the resulting protein curve is fit to the observed trajectories, assuming normal measurement errors with a hyperparameter for the standard deviation (σ), which constitute an additional model parameter to be inferred. Fitting was repeated 10 times per trajectory with Latin hypercube-sampled initial

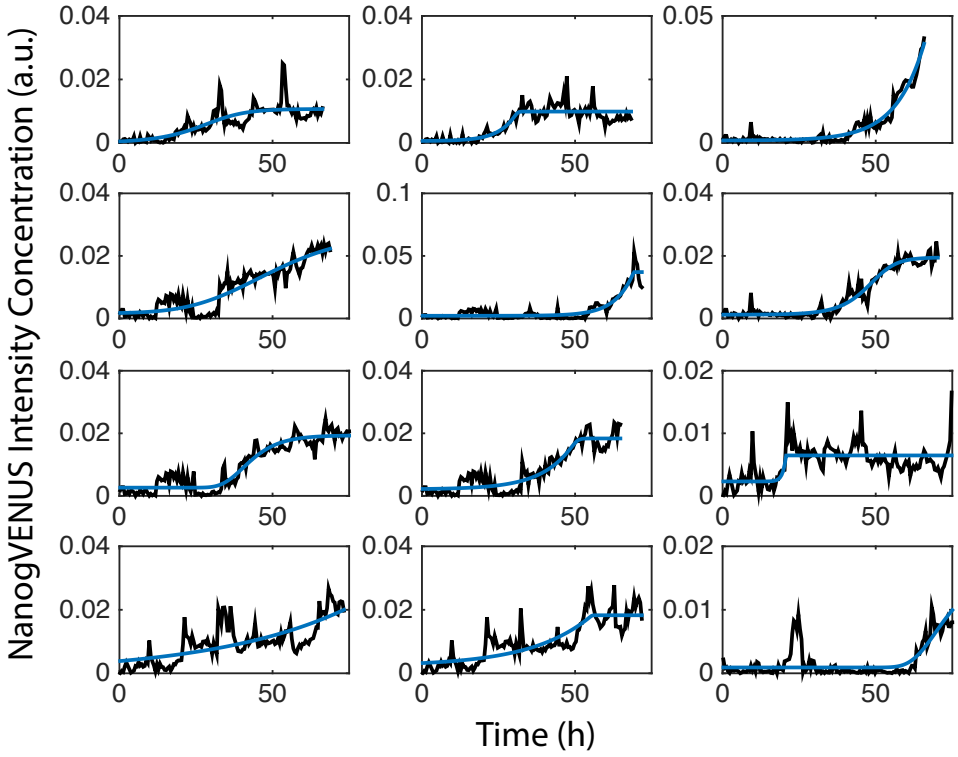

Figure 6.14: Example onsets of NanogVENUS intensity computed from one experiment (black), along with fit using a generalized logistic function (blue).

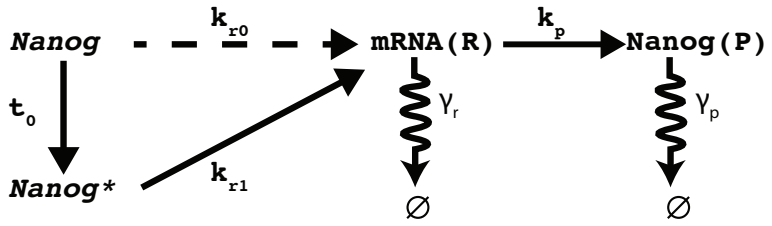

Figure 6.15: Switch model of NanogVENUS expression. NanogVENUS begins in the "inactive" configuration (Nanog) and switches to the active configuration (Nanog*) at time t_0. The inactive configuration has basal transcription rate k_{r_0} and the active configuration transcription rate k_{r_1}. Protein is translated at rate k_p, and mRNA (R) and protein (P) are degraded at rates γ_r, and γ_p, respectively.

Parameter	Description	Units	Lower Bound	Upper bound
k_{r_0}	Basal transcription rate	$\frac{\text{mRNA}}{\text{h·pixel}}$	0.05	50
k_{r_1}	Active transcription rate	$\frac{\text{mRNA}}{\text{h·pixel}}$	1	200
k_p	Translation rate	$\frac{\text{protein}}{\text{h·mRNA·pixel}}$	25	1.25×10^5
γ_r	mRNA degradation	$\frac{1}{\text{h}}$	0.01	1.5
γ_p	Protein degradation	$\frac{1}{\text{h}}$	0.01	1.5
t_0	Time of onset	h	0.1	100
σ	Observation error	Protein	0.1	1000

Table 6.4: Onset model parameter constraints

parameter values, using the `fmincon` numerical optimization function in Matlab with analytically-computed gradients. Parameter constraints used are listed in Table 6.4 and were chosen e.g. to be consistent with the NanogVENUS mRNA and protein half-lives of approximately 5h, or a degradation rate of 0.2 h^{-1}. Protein time series were normalized by the estimated nuclear area, as for the generalized logistic model above. Intensities concentrations (I) were then converted to absolute protein numbers (P) by multiplying by a rough estimate of $\lambda = 3.5 \times 10^5$ proteins/unit intensity, as approximated from a Western blot analysis, and by multiplying by the approximate average cell nuclear area of $\bar{A} = 200$ pixels: $P = \lambda I / \bar{A}$.

From each fitted trajectory, I estimated several "observable" features: the maximum intensity concentration level after onset (upper asymptote), the initial intensity concentration (lower asymptote), the time of onset, the time until half-maximum intensity level is reached $T_{1/2}$, and the maximum rate of production. I also compute the fraction of the cell cycle at which point the onset occurs, f, for the appropriate cell within the onset branch, see Figure 6.16. The statistics of the fitted curves are shown in Figure 6.17, and reveal both that onsets do not occur preferentially at any point in the cell cycle[1] (Figure 6.17C), and that onset times are very heterogeneous, occurring after as little as ten hours of imaging, but also as late as 60-70 hours (Figure 6.17B).

[1]Distribution does not differ significantly from the uniform distribution on $[0, 1]$, $p > 0.05$, Kolmogorov-Smirnoff test

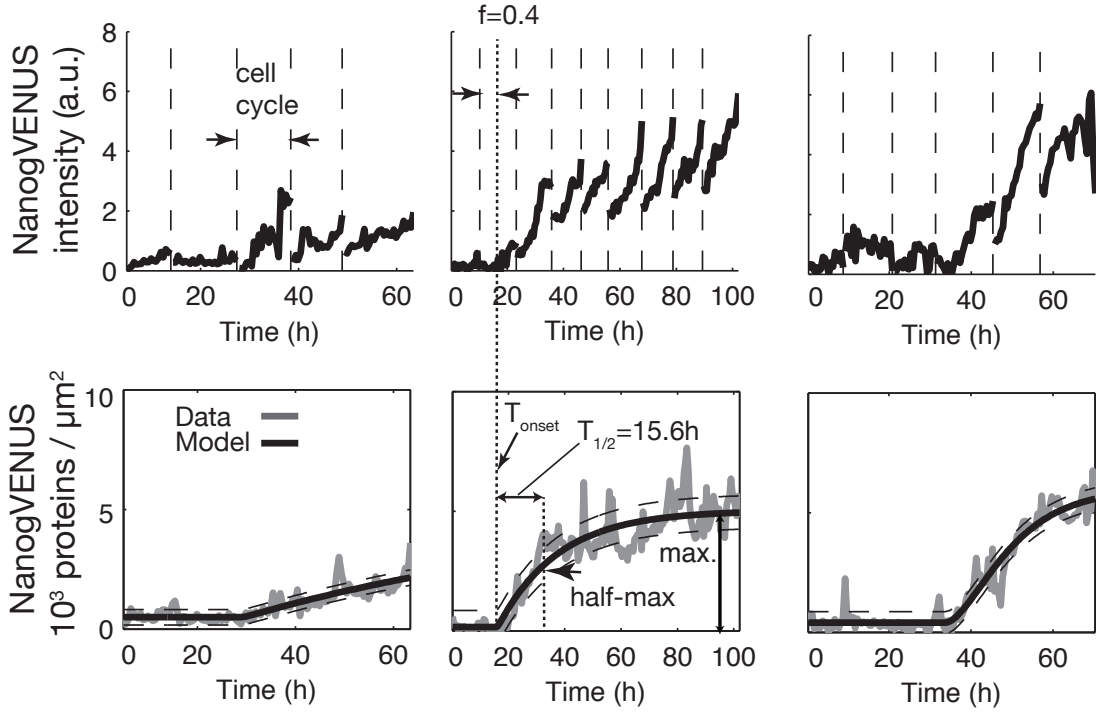

Figure 6.16: Three example onset trajectories computed by taking the NanogVENUS intensity trajectories (top row), and normalizing by the estimated nuclear volumes to obtain NanogVENUS intensity concentrations (bottom row). Each intensity concentration trajectory is fitted to the ODE model given in (6.2). From each fitted curve, we estimate the upper and lower asymptotes, the maximal rate of production, the time until half maximum intensity, and the time of onset. We further compute the fraction f within the cell cycle at which point the onset occurs.

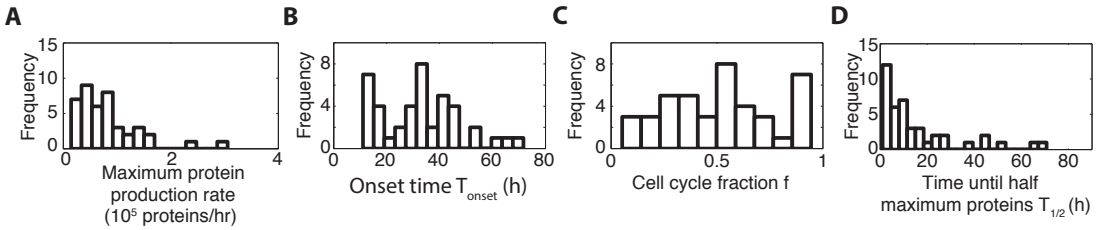

Figure 6.17: Distributions of observable features for the fitted onsets, as shown in Figure 6.16. A. The maximum protein production rate. B. The absolute time of NanogVENUS expression onset. C. The fraction within the cell cycle at which point the onset occurs. D. The time until half maximum proteins.

6.3.4 Memory

Variability among closely related cells

Next, we were interested in quantifying the extent to which the onset of NanogVENUS expression in populations of low-sorted mESCs was associated with cellular relatedness. Specifically, we were interested in analyzing whether sister cells, or more generally, cells of arbitrary degree of relation, show statistically significant correlation in their expression of NanogVENUS, and whether sister cells are more likely to both undergo onset of NanogVENUS expression than randomly chosen cells.

Using three low-sorted populations of mESCs, I computed the distribution of the differences in expression of several pluripotency factors between sister cells. Expression was measured via the quantification of immunohistological staining images using fluorescent antibodies against the pluripotency-associated transcription factors Oct4, Klf4 and Sox2, performed using QTFy; NanogVENUS intensity was quantified via fluorescence microscopy and QTFy. Only data for which the cells were fully tracked and for which endpoint staining was performed were utilized; in total 275 stem cells were analyzed.

The analysis revealed that sister cells, unsurprisingly, show decreased variability in the expression of pluripotency factors than randomly selected pairs of cells from the same experiment ($p = 0.0002641$ for Nanog, $p = 3.03 \times 10^{-11}$ for Oct4, $p = 5.063 \times 10^{-14}$ for Sox2, $p = 6.31 \times 10^{-8}$ for Klf4, Kolmogorov-Smirnoff test), see Figure 6.18. Variability was quantified using the standard error (the standard deviation normalized by the square root of the number of data points) of the quantified intensities of each biomarker. Thus, intensities that are more similar amongst closely related cells translates into a reduction in the standard error of each quantified factor.

However, for cousin cells (cells that had a common ancestor two generations prior), the NanogVENUS distribution is no longer significantly less variable than randomly chosen groups of four cells, see Figure 6.19. In contrast, the remaining factors Oct4, Sox2 and Klf4, show significantly decreased variability for cells related by a common ancestor even three generations prior, as compared to randomly selected groups of 8 cells ($p < 0.001$).

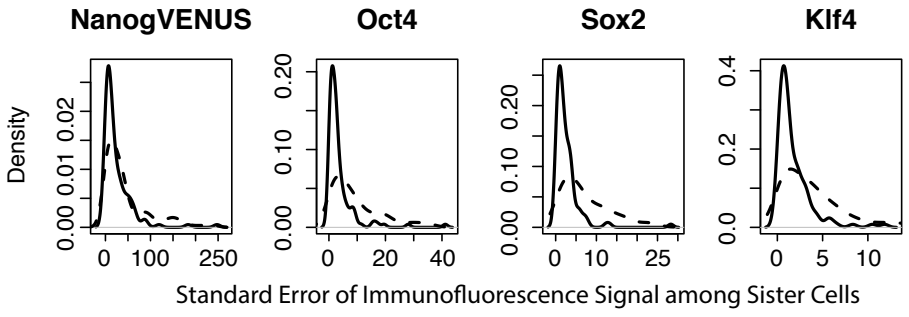

Figure 6.18: Sister cells (solid lines) exhibit significantly less difference in pluripotency factor expression as measured by immunofluorescence, as compared to random pairs of cells (dashed lines).

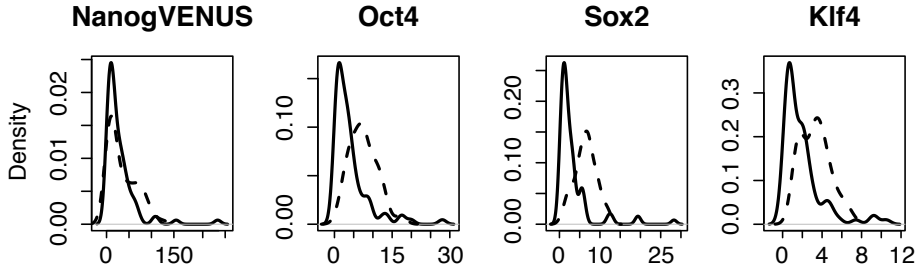

Figure 6.19: ES cells having a common ancestor up to three generations prior show significantly decreased variability in the expression of Oct4, Klf4, and Sox2 ($p < 0.001$, Kolmogorov-Smirnoff test), as compared to randomly selected groups of cells of the same size. NanogVENUS variability, however, is not significantly reduced compared to randomly selected groups of cells beyond the generation of division.

.

Congruent and incongruent divergence

Having determined that pluripotency expression remains less variable amongst cells that shared a common ancestor up to three generations prior, we next examined the extent to which the differential expression of NanogVENUS correlates with differential expression of the remaining pluripotency factors. To this end, I utilized a different set of mESC data that was fully tracked, inspected and quantified, along with endpoint stainings for the same factors Oct4, Sox2, and Klf4. Endpoint stainings were performed after 24h of continual imaging, for a total of 119 pairs of sister cells (see Table 6.1).

I then classified each pair of sister cells based on the relative expression of NanogVENUS in the pair. Pairs for which the expression of NanogVENUS was double or greater in one sister cell than in the other, or for which one sister cell showed detectable NanogVENUS signal while the other remained below the detection limit, were classified as "divergent", see Figure 6.20. The remaining pluripotency factors were subsequently categorized relative to the NanogVENUS expression—if a pair of cells exhibited a two-fold difference in the expression of a pluripotency factor it was classified as divergent, otherwise it was non-divergent. Furthermore, if a pair of cells divergent for a particular factor showed higher expression of that factor in the same sister cell for which the NanogVENUS expression was higher, then the divergence was termed "congruent divergence", indicating that the divergence was in the same fashion as for NanogVENUS; otherwise, the divergence was termed "incongruent", see Figure 6.21.

Examination of the sister cells for which NanogVENUS was divergent revealed that only Oct4 and Klf4 expression was significantly increased in the "high" cell—the cell that showed higher NanogVENUS expression, see Figure 6.22; Sox2 expression, while higher for the NanogVENUS high cell, was not statistically significant.

Using a multinomial distribution, I estimated the fraction of instances of each possible combination of pairs of divergent factors, that is, the frequencies for congruent, and incongruent divergence and non-divergence of each factor for the Nanog divergent sister cell pairs, and for the Oct4 divergent sister cell pairs, etc, see Figure 6.23. Interestingly,

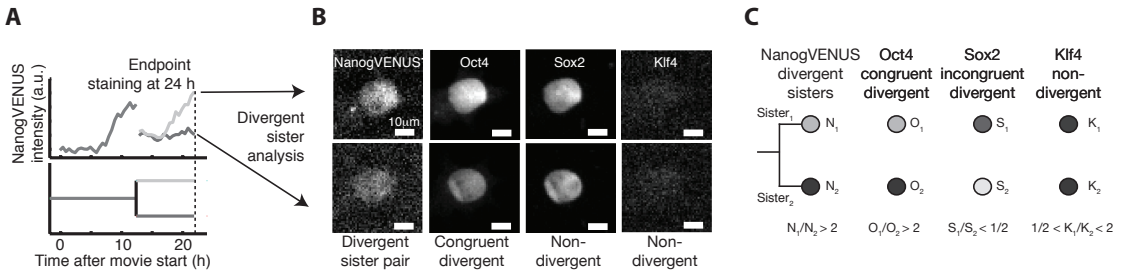

Figure 6.20: Endpoint staining of Oct4, Sox2 and Klf4 reveals congruent and incongruent divergence or non-divergence between sister cells, with respect to NanogVENUS divergence. A. Low-sorted mESCs are imaged for 24h, and endpoint staining is performed for Oct4, Sox2 and Klf4. B. Sister cell pairs for which NanogVENUS expression differs by two-fold or greater are deemed divergent for NanogVENUS. The remaining pluripotency factors are categorized as congruent or incongruent divergent, or non-divergent, depending on the ratio of expression of each factor between sister cells. C. Congruent divergence is defined as divergent (ratio of expression between sister cells two-fold or more), and relative expression levels the same ordering as for NanogVENUS (e.g. Oct4); incongruent divergence defined as divergent, but with the opposite ordering with respect to NanogVENUS (e.g. Sox2). Non-divergent is defined as expression ratio between sister of between 1/2 and 2 (e.g. Klf4).

NanogVENUS divergence seems to bear no relevance for divergence (congruent or incongruent) of the remaining factors, as evidenced by the high proportion of non-divergence (see Figure 6.23A). Conversely, the remaining factors show a high proportion of congruent divergence, indicating a likely coregulation of the pluripotency factors Oct4, Sox2, and Klf4, in agreement with the canonical core regulatory network of mESCs, see e.g. [8].

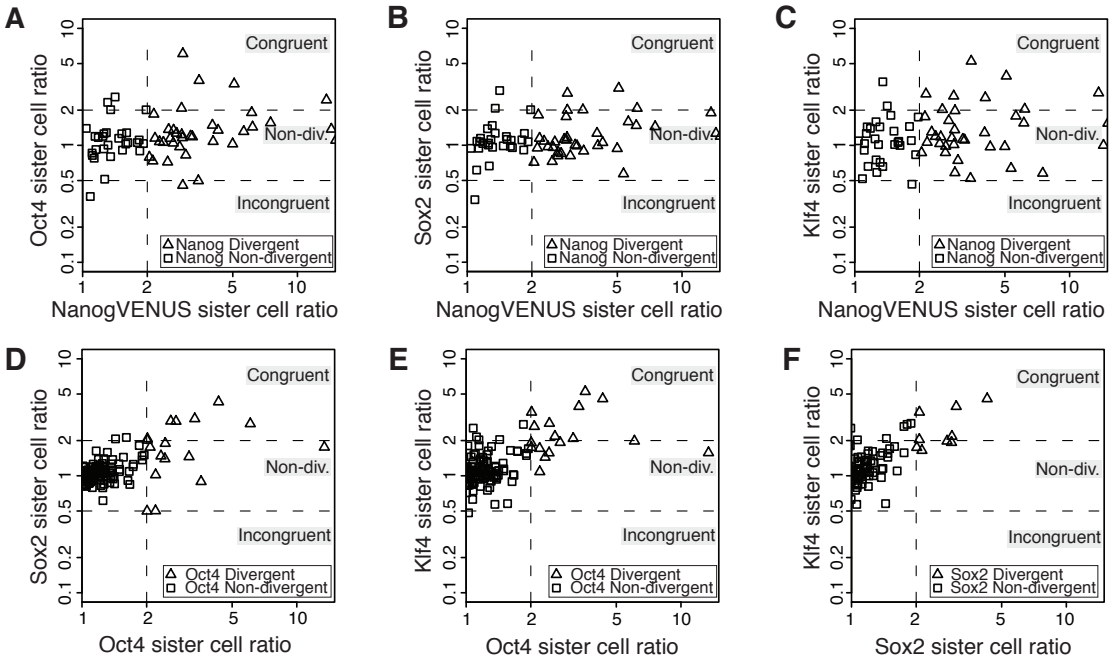

Figure 6.21: Sister cell pairs are categorized on the basis of the relative expression levels of transcription factors between sisters. If the expression of one sister exceeds the other by a factor of two or more the pair is deemed divergent. Other transcription factors for the same pair can then be congruently divergent if the same sister cell shows two-fold increased expression for that factor, incongruently divergent if the pair is divergent with inverted order, or non-divergent if the expression levels do not exceed the divergence threshold. The pairwise relationship of transcription factors among sister cell pairs is shown for each combination of transcription factors.

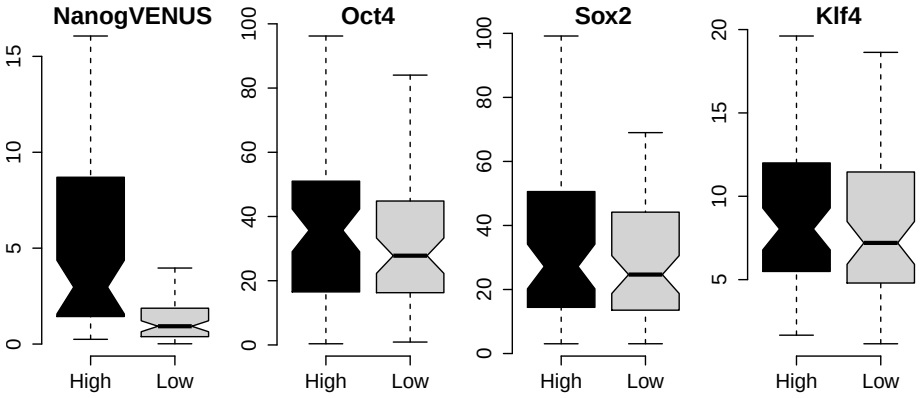

Figure 6.22: Pairs of sister cells for which the two cells diverge with respect to NanogVENUS signal shows significantly increased expression of Oct4 ($p = 0.001$), and Klf4 ($p = 0.025$), but not of Sox2 ($p > 0.05$). Pluripotency factor expression is visualized using box-and-whisker plots, where the whiskers extend to the maximum and minimum expression levels, the box extends to the 25% and 75% quantiles and the notch indicates the group median. Significance was determined using the non-parametric, paired, one-sided Wilcoxon signed rank test.

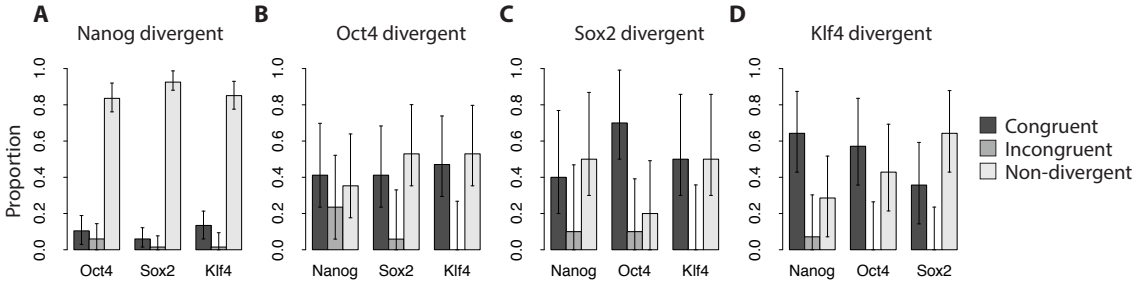

Figure 6.23: Sisters cell pairs can be classified according to the relative expression of transcription factors between sister cells, shown for 119 sister cell pairs from 4, 24h endpoint staining experiments. A. Sister cell pairs divergent for NanogVENUS are mostly non-divergent for other transcription factors. Sister cells divergent for other transcription factors show a high degree of congruence with the remaining factors for each of Oct4 (B), Sox2 (C) and Klf4 (D). Error bars estimated using a multinomial distribution for counts of each grouping, showing 95% confidence interval of the proportion.

6.4 Identification of subpopulations

Visualization of NanogVENUS dynamics in mESC colonies reveals an obvious heterogeneity in the behaviors of individual stem cells with some cells and lineages increasing in expression over time, some decreasing, some showing onset behavior, etc., see Figure 6.12. Hence, it is reasonable to assume that the mESC colonies might comprise several subpopulations, for which the dynamic evolution of the system differs.

By examining the estimated mother-daughter NanogVENUS intensity transition kernels (estimated using KDEs), for each of the low-sorted mESC colonies individually, it becomes clear that different trees exhibit different propensities for upregulation or onset, see Figure 6.24. For instance low-sorted trees 1, 3 and 9 all seem to only produce low-low transitions, i.e. the log-intensity of the daughter cell never rise above 0; in contrast the remaining trees show daughter log-intensities of up to 0.5 or 1.0, with an apparent attractor around daughter intensity of 0.0, showing a preference for increased NanogVENUS expression. This is confirmed by inspection of the trees, shown using absolute (non log-transformed) NanogVENUS intensities, see Figure 6.25, in which the same trees are shown to remain relatively low in NanogVENUS expression.

At this point the origin of the observed heterogeneous behavior of mESC colonies is unclear. It has been previously established that low-sorted mESC colonies are capable of reestablishing their steady state protein distributions. Thus the fact that some observed colonies undergo onset of NanogVENUS expression leading to higher expression levels and ultimately to reestablishing the steady-state is not surprising. It is clear also from the observation of individual colonies as in Figure 6.25, that some colonies remain low for several generations, which has not been previously observed. Indeed the predominant hypothesis has been that high NanogVENUS expression represents the "ground state", and thus that low expression is a transient phenomenon, giving rise to increased differentiation propensity. Hence, it is surprising to find colonies that remain low for NanogVENUS expression for sustained periods of time, in contradiction e.g. to the excitatory hypothesis [25], see Section 1.4.

In our subsequent investigations, we termed colonies which show sustained decreased NanogVENUS expression "Nanog Negative/low" colonies, and colonies which show increased expression "Nanog Mosaic" colonies, since they may contain a "mosaic" of cells with low and high NanogVENUS expression. With this distinction, we investigated whether the different colony types differ in other biologically meaningful ways, beyond the differential expression of NanogVENUS.

We performed time-lapse microscopy experiments for colonies of low-sorted mESCs, over a period of 4 days with continual imaging of NanogVENUS fluorescence intensity, and with an endpoint immunofluorescence staining for the pluripotency factors Oct4, Sox2 and Klf4, see Figure 6.26. We quantified the expression within colonies of the two types using QTFy to obtain single-cell expression levels for each pluripotency factor, see Figure 6.27, and for each combination of factors, see Figure 6.28. In each case, the pluripotency factors other than Nanog were expressed less, especially for Klf4 which shows the greatest difference between colony types. However, all other pluripotency factors were still expressed at detectable levels, within the NanogVENUS Negative/low colonies.

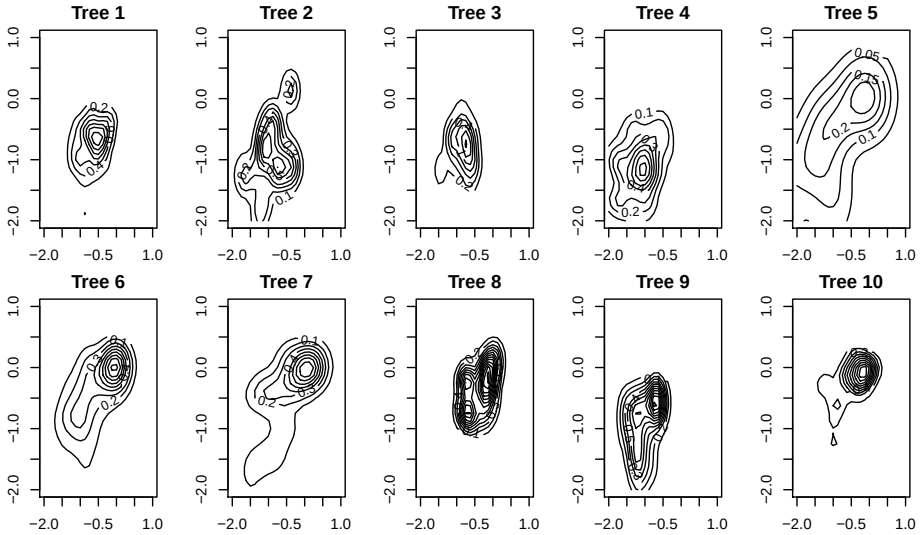

Figure 6.24: Low-sorted mESC colonies differ with respect to their mother-daughter NanogVENUS intensity transition kernels.

.

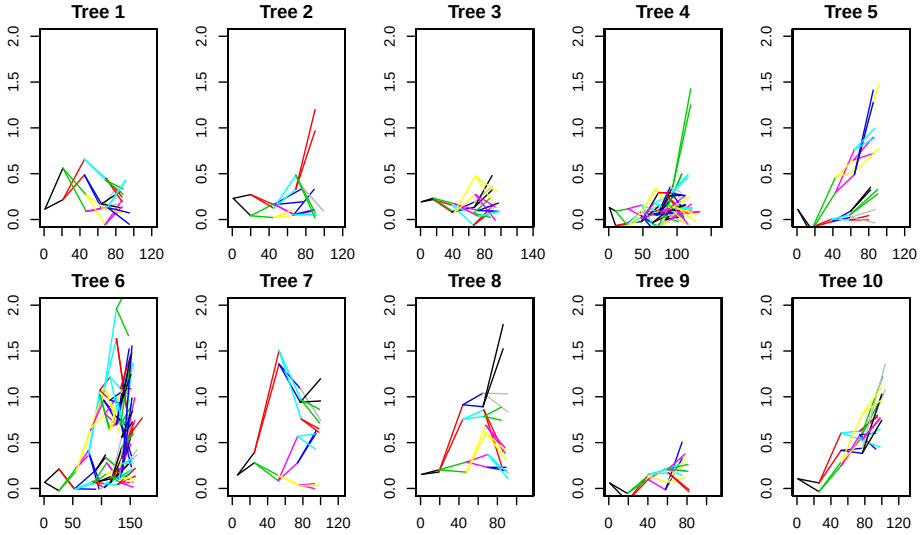

Figure 6.25: Low-sorted mESC colonies exhibit heterogeneous behaviors, in some cases remaining low in NanogVENUS intensity, and in other cases showing onset of NanogVENUS expression. Each cell is distinguished using a random color.

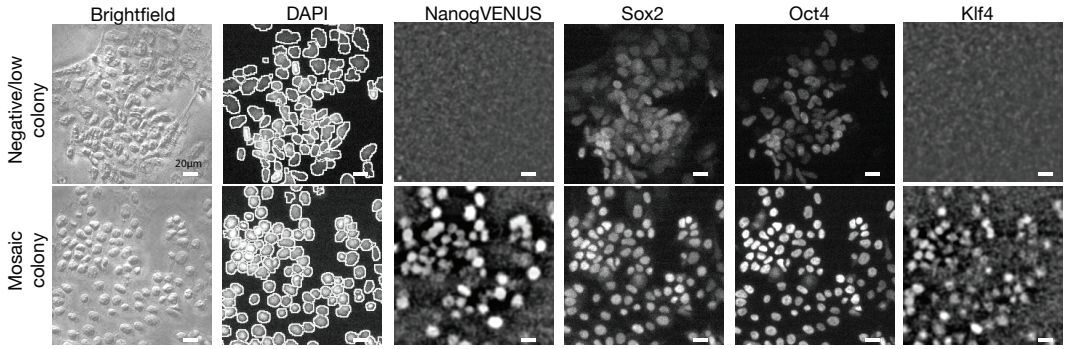

Figure 6.26: Low-sorted mESC colonies give rise to two phenotypically-distinct colony types, which differ in the expression levels of NanogVENUS. NanogVENUS Negative/low colonies how little or no NanogVENUS expression, and reduced levels of other pluripotency factors. NanogVENUS mosaic colonies contain cells with significantly higher NanogVENUS expression than any cell of the Negative/low colonies, and relatively higher levels of the remaining pluripotency factors.

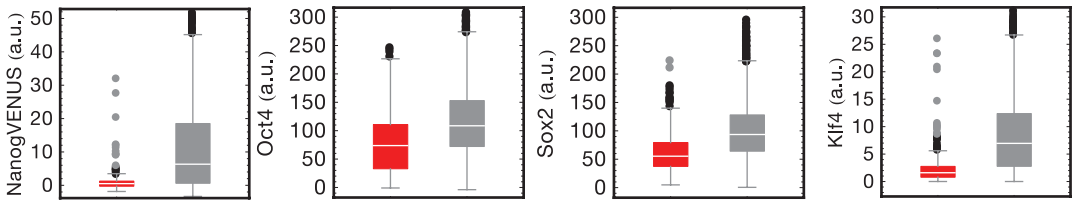

Figure 6.27: NanogVENUS mosaic colonies (gray) show increased expression levels for each pluripotency factor relative to the NanogVENUS Negative/low colonies (red).

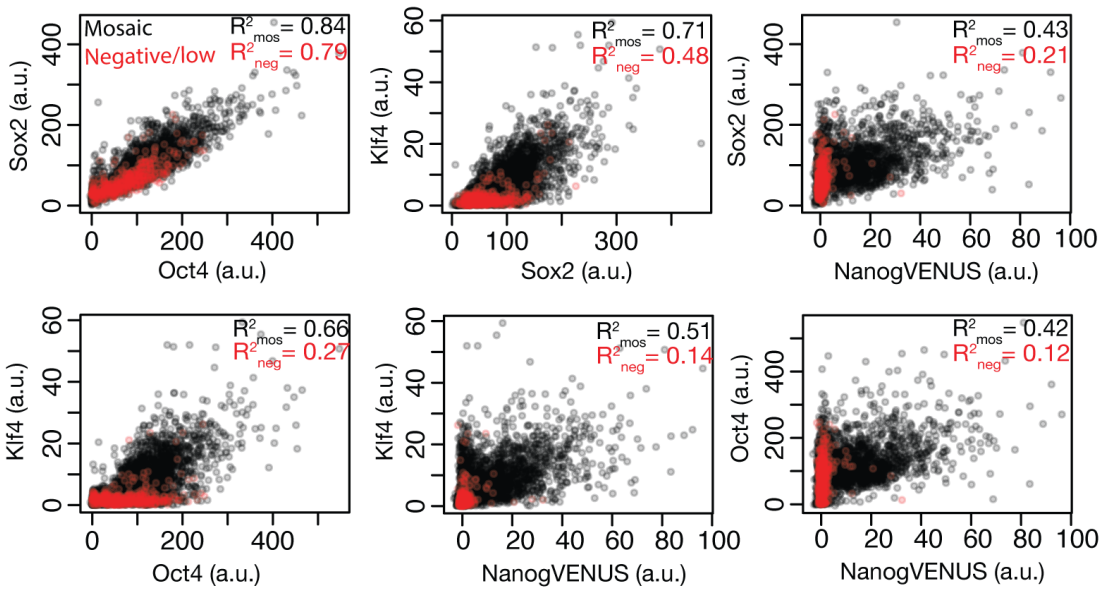

Figure 6.28: Both NanogVENUS mosaic (black) and Negative/low colonies (red) show high correlation between pluripotency factors. However, correlation between NanogVENUS and other pluripotency factors is reduced in mosaic colonies; NanogVENUS is very low or absent in NanogVENUS Negative/low colonies.

Correlation networks

Next, we attempted to identify possible differential regulation motifs by analyzing the relationship amongst the measured pluripotency factors in the two colony types. Specifically, we computed the Pearson and partial correlation networks (see Section 2.2.4), for each colony type, see Figure 6.29 (top). Interestingly, we find that the two colony types exhibit differential regulation, as determined by the different graph structure resultant when retaining only edges corresponding to statistically significant correlations ($p < 0.01$). Computing the partial correlation has the advantage of removing indirect interactions and can thus reveal correlations that might otherwise be "masked" due a strong, indirect interaction. Concretely, the partial correlation analysis reveals that the Nanog Negative/low colonies show a qualitatively different partial correlation between the factors Oct4 and Klf4 than do the mosaic colonies— significantly negatively partially correlated in Negative/low colonies, and significantly positively partially correlated in mosaic colonies. Moreover, by computing the (partial) correlation networks of the cells in the mosaic colonies for which the expression of NanogVENUS is no greater than the expression within the NanogVENUS Negative/low colonies ("low cells in mosaic colonies"), we find that the correlation networks more resemble those of the mosaic colonies and not those of the Negative/low colonies, suggesting that cells from mosaic colonies are phenotypically different as reflected in their correlation networks. The differences between the partial correlation networks of the two colony types were further confirmed by analyzing three biological replicates, see Figure 6.30, for which the Oct4-Klf4 partial correlation differs significantly between the colony types (Wilcoxon pairwise signed rank test, $p < 0.01$), and for which

the correlation is of opposite sign between the two types; the Oct4-Sox2 partial correlation also differs significantly.

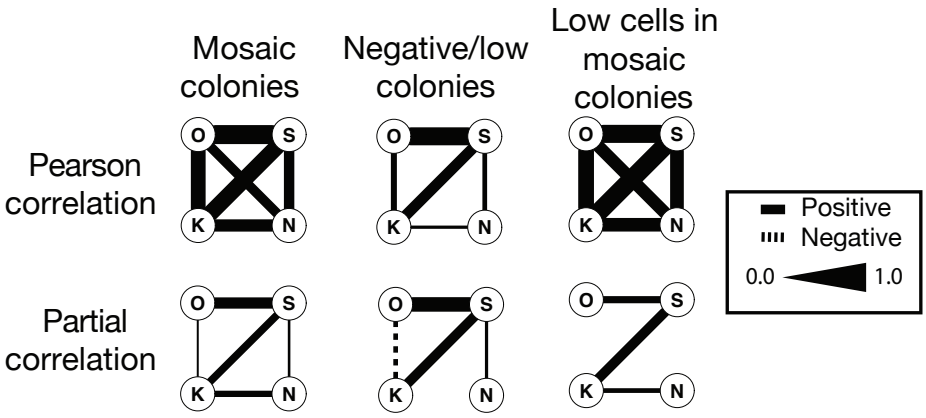

Figure 6.29: NanogVENUS Negative/low and Mosaic and differ with respect to their correlation networks of the pluripotency factors Nanog(N), Oct4(O), Sox2(S), and Klf4(K). NanogVENUS Mosaic and Negative/low colonies show significant positive correlation between each pluripotency factor when computing the Pearson correlation, with the exception of Oct4 / Nanog in the Negative/low colonies, which is not significant. However, when compared using partial correlations, NanogVENUS Negative/low and Mosaic colonies differ with respect to the Oct4 / Klf4 correlation: Mosaic colonies show positive positive correlation, whereas Negative/low significant negative partial correlation. Interestingly, cells in mosaic colonies with expression levels below similar to the Negative/low colonies ("low cells in mosaic colonies") show similar correlations and partial correlations to the mosaic colonies, suggesting that Nanog alone is not sufficient to predict the correlation networks colonies. Edge widths correspond to the magnitude of the correlations coefficient. Solid lines indicate positive correlations, and dashed lines negative.

Lastly, we examined the NanogVENUS mosaic and Negative/low colonies using the Multiresolution Correlation Analysis (MCA) method presented in Chapter 3. Interestingly, we find that all Nanog-sorted subpopulations within the mosaic colonies (including the low cells with NanogVENUS levels equal to the Negative/low colonies) show a positive Oct4-Klf4 partial correlation, whereas all Nanog-sorted subpopulations within the NanogVENUS Negative/low colonies show a negative Oct4-Klf4 partial correlation, see Figure 6.31, providing further evidence that this is a phenotypically distinct subtype, possibly with a distinct regulation mechanism.

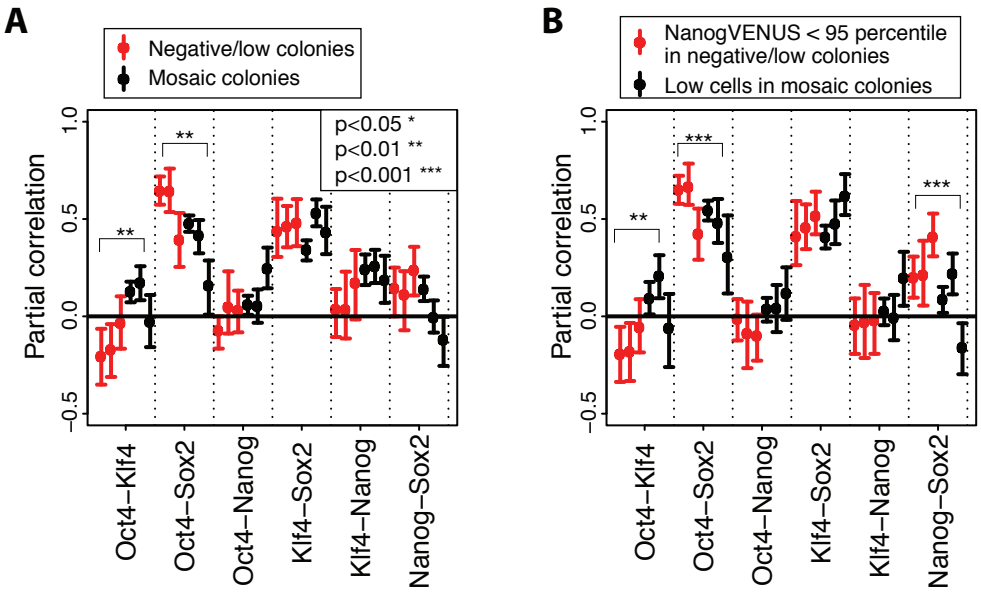

Figure 6.30: A. Comparison of partial correlations of transcription factor pairs reveals differences between Nanog mosaic and Nanog negative/low colonies. The most drastic difference occurs for Oct4-Klf4, where the partial correlation changes significantly from -0.14 ± 0.09 (mean $\pm$ SD, nE=3, nC=894, $p < 0.01$ using the pairwise Wilcoxon signed rank test to compare partial correlations between colony types for each replicate) in Nanog negative/low colonies to 0.09 ± 0.11 (nE=3, nC=3323) in Nanog mosaic colonies. We show mean and 95% bootstrap confidence interval for each replicate. Significant differences also appear for Oct4-Sox2, with 0.56 ± 0.14 and 0.35 ± 0.17 for Nanog negative/low and mosaic colonies, respectively. B. Nanog negative/low cells from Nanog mosaic colonies that express NanogVENUS at the same level as Nanog negative colonies still exhibit altered partial correlations. Significant differences are found for Oct4-Klf4, Oct4-Sox2, and Nanog-Sox2.

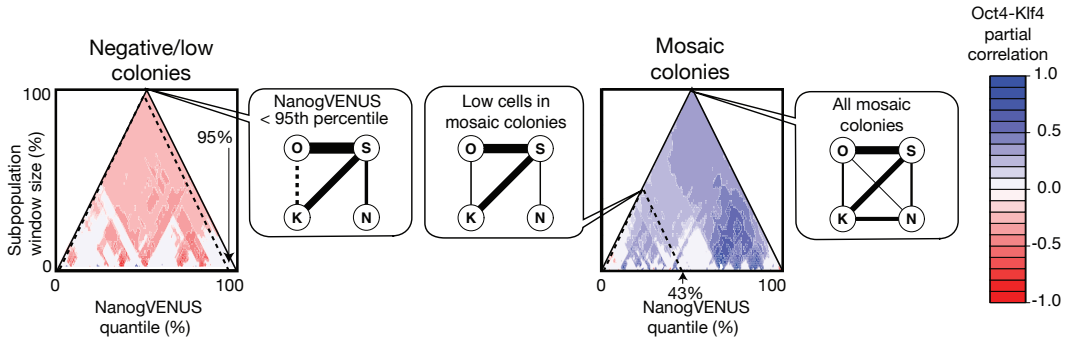

Figure 6.31: Multiresolution Correlation Analysis plots reveal that all Nanog-sorted subpopulations of the NanogVENUS Negative/low colonies (left) have either positive (red) or insignificant (transparent) Oct4-Klf4 partial correlations. In contrast, NanogVENUS mosaic colonies (right) show only positive (blue) or insignificant Oct4-Klf4 partial correlation. Cells in the mosaic colonies with NanogVENUS expression below the 95-percentile of the expression of the Negative/low colonies also show significant positive Oct4-Klf4 partial correlation. Complete pluripotency factor partial correlation networks for all Negative/low, and mosaic colonies, and for low cells within mosaic colonies are shown in callouts.

6.5 Stochastic auto-regulatory models for NanogVENUS dynamics

The analyses presented in the previous sections are largely descriptive, and provide insight into the transition rates between intensity levels in NanogVENUS mESCs, oscillations, subpopulations, and transitions from low/negative expression to higher NanogVENUS expression levels (onsets). However, the true mechanism regulating NanogVENUS expression is as of yet unknown. In particular, three motifs of Nanog transcriptional autoregulation have been hypothesized: no regulation, and negative or positive feedback. Interestingly, and somewhat perplexingly, all three models have been reported in the literature. For example, Ochiai *et al.* surmise that Nanog transcriptional activity is best explained by a random telegraph, with no regulation [147]. In contrast, most models assume positive autoregulation, see e.g. Refs. [25, 29, 146, 149, 156]. Lastly, two recent publications suggest Nanog negative autoregulation [22, 23]. Thus the real mode of autoregulation remains controversial. However, the NanogVENUS highly resolved time lapse fluorescence microscopy data presented in the previous sections might be able to afford additional insight into the true regulatory mechanism, when fit with stochastic gene regulation models.

The NanogVENUS data set obtained from the time lapse fluorescence microscopy experiments shown represent noisy, discrete, partially observed samples from a stochastic (biochemical) process, and thus are well suited to the Bayesian, tree-based inference algorithm presented in Chapter 5. With this algorithm, I perform parameter inference and model comparison for several candidate models of Nanog autoregulation, in order to assess the ability of these simple models to explain the observed time series. To utilize the particle filtering algorithm presented, I first converted intensities to absolute protein numbers using the rough conversion factor $\lambda = 3.5 \times 10^5$ proteins per unit fluorescence intensity,

obtained by Schwarzfischer *et al.* [236] (see Section 6.3.3). With this conversion factor, I derived protein copy number time series for a single fully-inspected NanogVENUS cellular genealogy, shown in Figure 6.32.

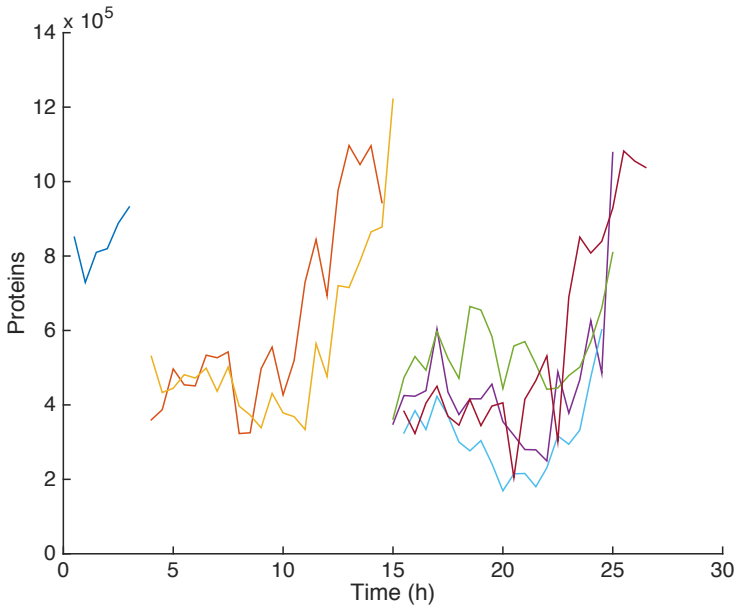

Figure 6.32: A low-sorted NanogVENUS genealogy emerging from a single progenitor cell. Individual cells are colored for ease of visualization.

Based on the magnitude of the fluctuations of the fluorescence signal, I estimated the standard deviation of the measurement error to be approximately 10^5 molecules. For the stochastic gene regulatory model generating the data, I consider the three candidate transcriptional autoregulatory motifs analyzed in Chapter 5: no feedback, negative feedback, and positive feedback. For inference, I used the prior distributions for model parameters shown in Table 6.6 (mean and standard deviations shown in Table 6.7). The observed NanogVENUS signals varied from approximately $10^5 - 10^6$ proteins (see Figure 6.32). Thus the squared number of proteins is approximately $10^{10} - 10^{12}$. By choosing prior distributions for the on and off rate constants in the Positive and Negative Feedback models with a mean of 10^{-11}, the expected waiting time for the on and off switches is thus approximately between $0.1 - 100$ hours, which should thus lead to several switches in the simulated time series. Moreover, for time series with suitably small k_{on} values, the rate of DNA activation should initially be relatively small but increasing substantially with the protein copy number in the Positive Feedback model, and similarly for the DNA inactivation rate in the Negative Feedback case.

Computing the marginal likelihoods (see (5.8)) I find that the No Feedback model and Negative Feedback models are preferred to the Positive Feedback model with mean Bayes Factors of 445.8 and 200.3, respectively, see Table 6.5. However, the No Feedback model is only weakly preferred to the Negative Feedback model with a Bayes Factor of 2.22.

Replicate	No Feedback	Negative Feedback	Positive Feedback
1	-1810.2	-1811.0	-1817.1
2	-1810.2	-1810.8	-1815.3
3	-1810.3	-1811.3	-1816.6

Table 6.5: Marginal log likelihoods of each model for the NanogVENUS tree.

	+ Feedback		- Feedback		$\emptyset$ Feedback	
Parameter	α	β	α	β	α	β
k_{on}	1	10^{11}	2	2	2	2
k_{off}	2	2	1	10^{11}	2	2
k_m	10	0.1	10	0.1	10	0.1
d_m	2	10	2	10	2	10
k_p	3	0.0005	3	0.0005	3	0.0005
d_p	2	10	2	10	2	10

Table 6.6: Gamma prior distribution shape parameters used for NanogVENUS inference for the Positive (+), Negative (-), and No Feedback ($\emptyset$) models. Parameter descriptions and units are the same as in Table 5.3.

Interestingly, the marginal log likelihoods of each model are very consistent over each of the three replicates.

I next examined the posteriors distributions of the model parameters for both the No Feedback model and the Negative Feedback model. For both models, I find that the posterior converges to a very similar distribution in each replicate of the inference algorithm. In each case, I computed the posterior of each replicate and the average posterior over the replicates, obtained by resampling with equal probability from each of the three posteriors. For the No Feedback model all parameters except k_{off} show a visible shift from the prior distribution, see Figure 6.33. Interestingly, I find that the parameters for the protein birth death process k_p and d_p show fairly noisy posterior distributions, whereas the estimates for k_{on} and k_{off} are very smooth. This is consistent with the fact that the conditional probability density of each model parameter is a gamma distribution, with parameters that grow linearly with the number of reaction firings or the integral of the propensity function. In the case of protein, the copy numbers are drastically higher (on the order of $10^5 - 10^6$), as opposed to the DNA or mRNA copy numbers which are either 1 or up to a few hundred, respectively. Thus the number of birth and death reaction firings for the proteins are much higher than for the other species, leading to increasingly narrow posterior distributions for the protein model parameters. The narrow conditional distributions lead to the overall more "spiky" appearance for the protein parameters as compared to the DNA and mRNA parameters, for which the conditional distributions are considerably larger.

Examining the marginal posterior distributions of model parameters for the Negative Feedback model, I find that all model parameters except k_m show an obvious shift from the prior distribution, see Figure 6.34. In particular, the parameters $k_{\text{on}}, k_{\text{off}}$ and k_p are particularly well identified, with much narrower distributions than the prior with a high

	+ Feedback		- Feedback		∅ Feedback	
Parameter	mean	std	mean	std	mean	std
k_{on}	1e-11	1e-11	1	0.7	1	0.7
k_{off}	1	0.7	1e-11	1e-11	1	0.7
k_m	100	31.6	100	31.6	100	31.6
d_m	0.2	0.1	0.2	0.1	0.2	0.1
k_p	6000	3464.1	6000	3464.1	6000	3464.1
d_p	0.2	0.1	0.2	0.1	0.2	0.1

Table 6.7: Gamma prior means and standard deviations used for NanogVENUS inference.

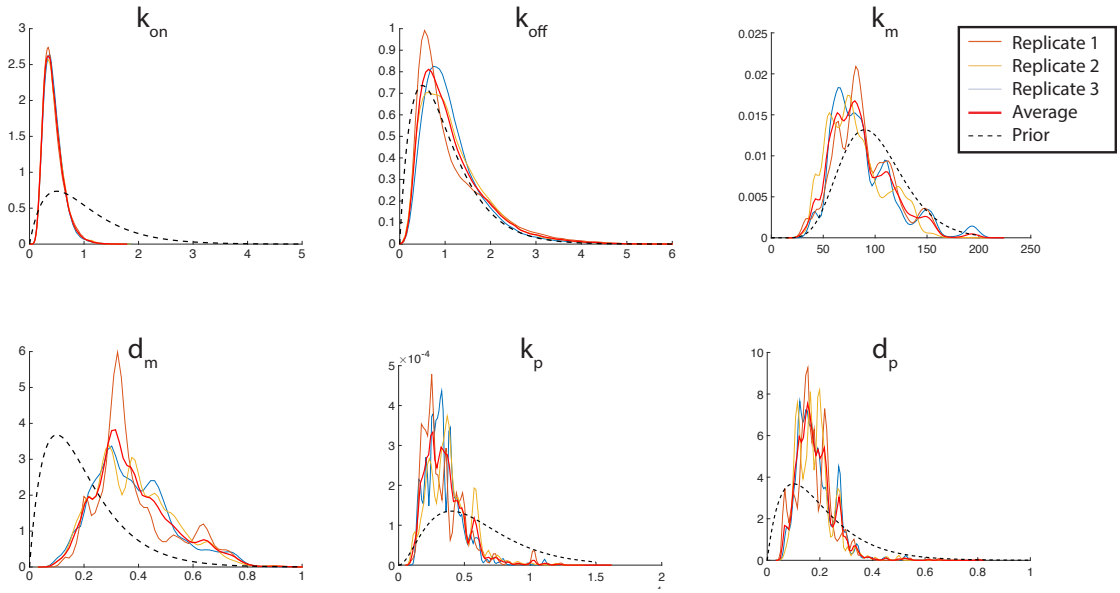

Figure 6.33: Posterior distributions of model parameters for the No Feedback model fitted to the NanogVENUS dataset shown in Figure 6.32, shown for three replicates of the inference procedure.

degree of consistency among replicates. In contrast, the transcription rate k_m shows less deviation from the prior than in the No Feedback case. Interestingly, the distributions of all model parameters except for those pertaining to the DNA activation and inactivation process are quite consistent between the No Feedback and Negative Feedback models.

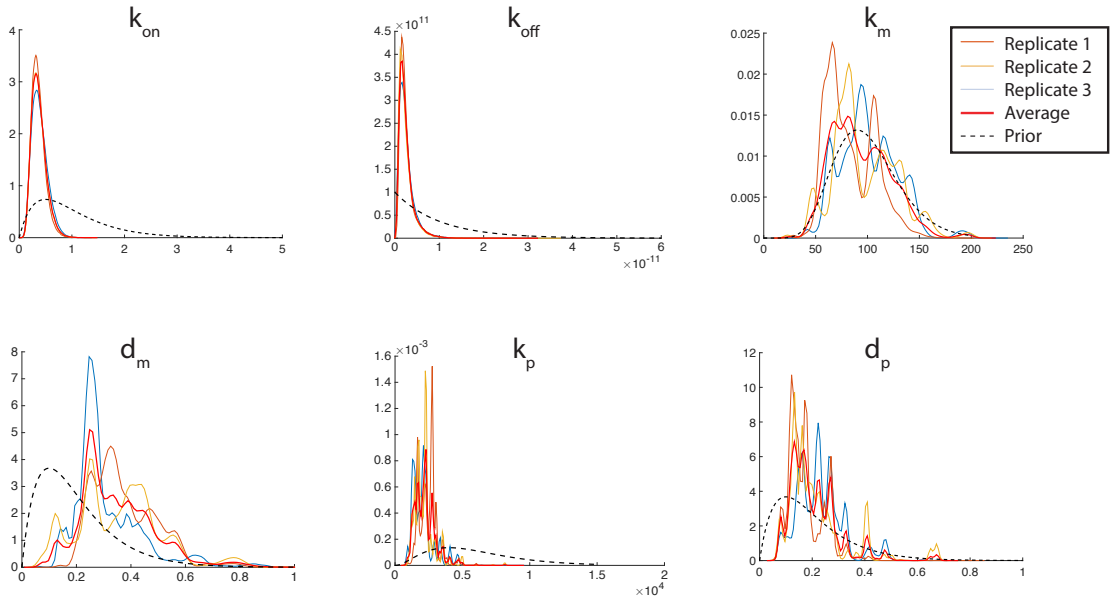

Figure 6.34: Posterior probability distributions of model parameters for the Negative Feedback model fitted to the NanogVENUS dataset shown in Figure 6.32, shown for three replicates of the inference procedure.

In summary, when applied to a real NanogVENUS tree dataset, I find that the Bayes Factors analysis provides definitive evidence against the Positive Feedback model. However, it appears that the high measurement error of the fluorescence signal renders it difficult to discriminate between the No Feedback and Negative Feedback models; the average likelihood of each transition in the two models was similar enough (i.e. the simulated trajectories were quite similar) that the Bayes Factor was not large enough to rule definitively in favor of one model. Thus in order to better discriminate between the models it may be necessary to either extend the inference procedure to deeper trees (the current analysis is limited to just 3 generations), improve the quality of the data via e.g. better background correction algorithms in the image processing steps, or repeat the analysis for multiple cellular genealogies and see if data accumulates in favor of a particular model. The Negative Feedback and No Feedback models yielded very similar results for the parameter inference (except for the DNA inactivation rate, where the models differ structurally), suggesting a robust estimation of the model parameters. Interestingly, the inference results suggest a lower protein production rate and higher mRNA turnover rate (half-life approximately 2-5 hours) than assumed in the parameter priors; for comparison, Ochiai *et al.* estimated a mRNA half-life of about 4.8h [147]. Both models also predict a DNA activation rate with an average waiting time of approximately 2-3 hours, suggesting

fairly rapid transitions between active and inactive states. For comparison, Ochiai *et al.* estimate an average waiting time of 0.6h in serum/LIF conditions using a transcriptional reporter. However, both Singer *et al.* [31] and Ochiai *et al.* report [147] evidence for the existence of two subpopulations, which differ in their relative Nanog transcription rates, suggesting that the slow transitions between active and inactive promoter conformations predicted by our inference framework might in part be due to switching between phenotypically varying subpopulations. This is also in line with our observation that low-sorted NanogVENUS mESCs are capable of generating low/negative and mosaic colonies with differing partial correlation networks and NanogVENUS expression dynamics. Thus, it would be interesting to see if stochastic models which explicitly incorporate switching of cells between "modes" with different reaction constants and/or CRN topologies are favored by model selection using Bayes Factors. This investigation and the application of the inference framework to other models is left for future work. We note however, that such a model would be amenable to the inference procedure of Chapter 5 through the inclusion of an additional stochastic "reaction" at the time of division.

Chapter 7

Discussion

Nanog is a key factor for the maintenance of pluripotency in mESCs and thus it is essential to study its expression dynamics in order to provide a complete understanding of the regulation of pluripotency and differentiation. In particular, a detailed mechanistic model of mESC regulation encompassing its key transcription factors would facilitate the quantitative prediction of the response of mESCs to perturbations such as knock-downs, spike-ins, etc. both deepening our understanding of fundamental stem cell biology and potentially paving the way for the rational design of protocols for directed differentiation of human ESCs for use in clinical applications.

In addition to being a key regulator of pluripotency in mESCS, Nanog is also known to be heterogeneously expressed, both at the transcriptional and translational levels, when cultured in serum/LIF. Moreover, mESCs with low Nanog expression have been shown to be at increased risk of differentiation, in agreement with Nanog's role as a sustainer of pluripotency. Nanog's heterogeneous expression has led to the proposal of many models capable of giving rise to bimodal distributions in approximate agreement with the observed Nanog distributions. Such models, however, largely rely on *ad hoc* model assumptions pertaining to model structure and the exact form of the equations describing the dynamical systems capturing protein dynamics. In the literature, the emphasis has been on inventing models that give rise to bimodality, via a variety of mechanisms including stochastic oscillations, excitations, bistablity emergent from positive feedback loops or stochastic switching between cell states with high and low Nanog protein production rates. Such models are tuned to qualitatively match the observed bimodal distribution of the Nanog steady state, but have not been fit to real Nanog time-courses due in part to the lack of availability of quantitative, time-resolved data. Furthermore, no attempt has been made previously to systematically identify the best model for describing real Nanog protein dynamics.

In this work, I analyzed Nanog protein time-courses generated using fluorescence time-lapse microscopy of colonies of mESCs containing the NanogVENUS fluorescent reporter knock-in for Nanog. In particular, in Chapter 6, I described the transitions observed between mother and daughter cells using a Markov model for populations of cells obtained either by sorting for high or low NanogVENUS expression, or without sorting. Investigations revealed that both oscillations and excitatory dynamics are unlikely given the observed transitions, casting doubt on previously hypothesized models. I further modeled

the onset of NanogVENUS expression in low-sorted mESC colonies, using a simple ODE model with a random switching event. Using this approach, I concluded that onsets occur with very heterogeneous onset dynamics, and I showed in particular that there is no inherent preference for the relative time of the onset within the cell cycle.

Using the NanogVENUS onset time-courses, we identify two types of colonies emerging from low-sorted mESCs, corresponding to NanogVENUS Negative/low expression, and to NanogVENUS mosaic expression, where the latter contains a mixture of cells with both non-detectable or low NanogVENUS expression and cells with mid-to-high NanogVENUS expression. These populations were later assayed for differentiation potential, and it was revealed that the NanogVENUS Negative/low population tend to express Foxa2, an endodermal marker, at lower levels than the cells from mosaic colonies, upon exposure to retinoic acid, an inducer of differentiation toward the neuroectodermal lineage. Sox1, a neuroectodermal lineage marker, was also markedly decreased in the Negative/low cells [234]. I further investigated cells from the two colony types using multiresolution correlation analysis (MCA), a novel technique for visualizing the local correlation structure of low-dimensional datasets, see Chapter 3. Using MCA, I determined that the two subpopulations significantly differed with respect to their partial correlation networks, suggesting differential regulation.

In a first step towards inference of stochastic models for NanogVENUS dynamics, I investigated the utility of a recently developed approximation to the solution of the chemical master equation, based on geometric singular perturbation (GSP), for a two-stage model of gene regulation, see Chapter 4. The GSP approach improves upon a previous model by Shahrezaei *et al.* [111], which assumes infinite scale separation between mRNA and protein degradation rates—an assumption which may be justified in a prokaryotic context, but which does not hold for eukaryotes where e.g. mRNA and protein half-lives do not greatly differ. However, the GSP approach proved problematic for parameter inference due to difficulties evaluating the special functions arising in the computation to the very high numerical precisions needed for accurate evaluation of the probability densities. More importantly, since the method only is accurate to first order in the perturbation parameter, it frequently generates non-physical, negative transition densities between states. The presence of negative probabilities severely impeded the inference procedure, from which I concluded that the method, although useful for approximating both the transient and steady-state probability densities, is not suitable for parameter inference with time-lapse fluorescence data such as the NanogVENUS dataset. However, the method may become suitable with the inclusion of further terms of the perturbation series, in future work.

Lastly, in Chapter 5, I developed a fully-stochastic, exact Bayesian parameter inference framework for performing parameter inference and model selection using tree-structured, partially and discretely-observed protein time series data, as is the case for NanogVENUS. The method is based on the bootstrap (recursive) particle filter, which provides successive approximations of the posterior distribution of the model parameters for a given model topology, i.e. a chemical reaction network with fixed reactions. At each iteration of the algorithm, additional observed data points are included corresponding to the set of observations for that timepoint, and the posterior distribution of the parameters updates accordingly. I used synthetic data to demonstrate that the parameter inference procedure learns the correct model parameters for three simple transcriptional autoregulatory

models of gene expression, and that, when combined with Bayes Factors, the methodology correctly identifies the mode of regulation for each model. The algorithm extends the standard bootstrap particle filter by including an additional step for simulating cell division and the accompanying random partitioning of cell contents (mRNA and protein) to the two daughter cells. By including cell division, I was able to extend the inference to tree-structured data, corresponding e.g. to colonies of proliferating cells as is the case for the NanogVENUS mESC data analyzed. Lastly, in Chapter 6, I applied the tree-based inference methodology to a fully-inspected, three generation NanogVENUS cellular genealogy from a low-sorted mESC founder cell, and concluded that the model comparison provides strong evidence against the positive feedback model. However, with the noisy trajectories examined, it was not possible to definitively discern between the models with no feedback and negative feedback.

In future work, I will continue to investigate stochastic models of Nanog expression utilizing the algorithm presented in Chapter 5. In particular, I will apply the method to all available mESC NanogVENUS data on a colony-by-colony basis, in order to rigorously identify the stochastic models most in agreement with the data, with the aid of model comparison using Bayes Factors. As previously discussed, the inference framework is general enough to easily permit the comparison of arbitrary models such as the switching model proposed by Ochiai *et al.* [147], models with translational control, and others. Hence, I will perform a comparison of stochastic models for Nanog regulation, which will help identify candidate models to be tested with further biological experiments. The aims of this comprehensive analysis will be to identify the mechanisms underlying NanogVENUS genealogies, and to assess the robustness of estimated model parameters for biological replicates and across experimental conditions.

By analyzing NanogVENUS expression dynamics we have characterized the dynamics of the onset of gene activity, revealed that mESCs do not undergo stable oscillations in NanogVENUS expression, and identified the presence of two mESC colony subtypes differing in expression levels of key pluripotency factors, correlation between these factors and significantly different cell fate outcomes upon exposure to inducers of differentiation. The investigation of NanogVENUS dynamics in mESCS will be complemented by future investigations including additional fluorescent protein reporter knock-ins for pluripotency factors including Oct4 and Klf4. Much of the same analysis can be performed for two-color datasets, to investigate e.g. potential mESC subpopulations arising from heterogeneous pluripotency factor expression.

The following investigations are left for future work:

- Application of the tree-based particle filtering algorithm to more NanogVENUS datasets
- Investigation of a wide variety of models for NanogVENUS dynamics
- Embedding of the tree-based particle filter within an MCMC algorithm in order to increase efficiency and permit hyperparameters such as measurement error variance
- Development of a graphical user interface for the MCA method

In conclusion, the tools and results presented in this thesis will facilitate future research into the underlying regulatory mechanism of mESCs as well as other biological systems for

which single-cell data are available. A deeper understanding of mESC regulation, in turn, may provide valuable insight for corresponding regulation in human embryonic stem cells, or in human induced pluripotent stem cells which are rapidly gaining traction in tissue replacement therapies. Thus, this work contributes to the growing body of knowledge of the fundamental biology of mESCs and may facilitate future clinical applications.

Appendices

Appendix A

Common probability distributions

A.1 Normal distribution

One of the most commonly occurring distribution in many statistical problems is the **normal** distribution, or **Gaussian** distribution. In one dimension, it is described by the probability density function

$$\phi(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(x-\mu)^2}{2\sigma^2}) \tag{A.1}$$

where μ is the mean, and σ the standard deviation. The prefactor $\frac{1}{\sqrt{2\pi\sigma}}$ is necessary to ensure normalization, i.e. that the integral over the domain of the event space $(-\infty, \infty)$ is one. The extension to higher dimensions is given by

$$\phi(\mathbf{x};\boldsymbol{\mu},\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(x-\boldsymbol{\mu})) \tag{A.2}$$

where $\boldsymbol{\mu}$ is the (multivariate) mean vector, and Σ is the covariance matrix.

A.2 Log-normal distribution

The (univariate) **log-normal distribution** is defined by the probability density

$$\phi(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma x} \exp(-\frac{(\log x-\mu)^2}{2\sigma^2}) \tag{A.3}$$

By comparison to (A.1), it is easy to see that the natural logarithm of a random variable X is normally distributed if X is log-normally distributed.

A.3 Exponential distribution

Another common distribution is the **exponential distribution** which arises in the context of waiting times until an event occurs. If the probability per unit time of the event occurring is a constant λ, then one can compute the probability that the time τ that the event

first occurs in the infinitesimal time interval $[t, t+\Delta t]$, as $P(t \leq \tau \leq t+\Delta t) = \lambda \Delta t P(\tau \geq t)$. Using the cumulative probability density $\Phi(t) = \int_0^t \phi(s)ds$ this can be rewritten as

$$\Phi(t + \Delta t) - \Phi(t) = \lambda \Delta t (1 - \Phi(t)) \tag{A.4}$$

rearranging and taking the limit $\Delta t \to 0$, one obtains the ODE $\Phi'(t) = \lambda(1 - \Phi(t))$ which has solution $\Phi(t) = 1 - \exp(-\lambda t)$. Hence, the density of an exponential distribution with parameter λ is given by:

$$\phi(t) = \frac{d}{dt}\Phi(t) = \lambda \exp(-\lambda t). \tag{A.5}$$

Competing exponentials

Consider the case of $R > 1$ exponentially-distributed events occurring stochastically. The cumulative probability density that event i has occurred at or before time t is given by $\Phi_i(t) = 1 - \exp(-\lambda_i t)$. Hence, the probability of it *not* having occurred until time t is given by $1 - \Phi_i(t) = \exp(-\lambda_i t)$. The probability of *none* of the events $i = 1 \ldots R$ occurring until time t is then simply:

$$P(t_1 > t, t_2 > t, \ldots, t_R > t | \lambda_1, \ldots, \lambda_R) = \prod_{i=1}^{R} \exp(-\lambda_i t) = \exp\left(\sum_{i=1}^{R} -\lambda_i t\right). \tag{A.6}$$

Finally, the probability of any event taking place at time t, and no event before time t, is given by:

$$\begin{aligned} \phi(t) &= (\lambda_1 + \ldots + \lambda_R) \exp(-\lambda_0 t) \\ &= \lambda_0 \exp(-\lambda_0 t) \end{aligned} \tag{A.7}$$

where $\lambda_0 = \sum_{i=1}^{R} \lambda_i$. Thus, (A.7) is the probability density of the waiting time until any of the events $i = 1 \ldots R$ occurs. Given that one of the events occurred at time t, the probability that the i^{th} event occurred, and not another, is given by $P(t_1 > t, t_2 > t, \ldots, t_i = t, t_{i+1} > t, \ldots) = \lambda_i / \lambda_0$.

A.4 Poisson distribution

The Poisson distribution with parameter λ is defined by the density

$$\phi(x; \lambda) = \frac{\lambda^{-x} \exp(-\lambda)}{x!} \tag{A.8}$$

It plays an important role in the statistics of random events, such as the probability of a certain number of events occurring in some time interval if the probability per unit time is constant (i.e. an exponential process).

For an exponential process with parameter λ, the probability of the event occurring at time t is given by (A.5). Consider a time interval dt sufficiently small such that the probability of more than one event occurring is negligible. The probability of N events occurring until time $t + dt$ is thus the probability that $n - 1$ events occurred until time t,

and one event occurred thereafter, plus the probability that n events occurred until time t, and no events occurred thereafter, given by:

$$P(n, t+dt) = P(n-1, t)\lambda dt + P(n, t)(1 - \lambda dt) \tag{A.9}$$

hence

$$\begin{aligned} \frac{P(n, t+dt) - P(n, t)}{dt} &= P(n-1, t)\lambda - \lambda P(n, t) \\ &= -\lambda \left[P(n, t) - P(n-1, t) \right]. \end{aligned} \tag{A.10}$$

In the limit $dt \to 0$ this yields the recursive ODE $\dot{P}(n, t) = -\lambda(P(n, t) - P(n-1, t))$. Substituting (A.8) yields

$$\begin{aligned} \frac{d}{dt}\left[\frac{(-\lambda t)^n \exp(-\lambda t)}{n!}\right] &= \frac{\lambda n (\lambda t)^{n-1} \exp(-\lambda t)}{n!} + \frac{(\lambda t)^n (-\lambda) \exp(-\lambda t)}{n!} \\ &= -\lambda \left[\frac{(\lambda t)^n \exp(-\lambda t)}{n!} - \frac{(\lambda t)^{n-1} \exp(-\lambda t)}{(n-1)!}\right] \\ &= -\lambda \left[P(n, t) - P(n-1, t)\right] \end{aligned} \tag{A.11}$$

verifying (A.10).

A.5 Gamma distribution

The gamma distribution is a flexible distribution parametrized by two parameters α and β, known as the scale parameter and shape parameter, respectively. The gamma distribution can take on a variety of shapes depending on the choice of parameters, ranging from exponential-like distributions to peaked, normal-like distributions. The mean is given by $\frac{\alpha}{\beta}$ and the variance by $\frac{\alpha}{\beta^2}$. The probability density function is given by:

$$\phi(x; \alpha, \beta) = \frac{\beta^\alpha x^{\alpha-1} e^{-x\beta}}{\Gamma(\alpha)}. \tag{A.12}$$

A.6 Gauss hypergeometric distribution

The Gauss hypergeometric distribution, denoted ${}_2F_1(a, b; c; z)$ is defined as

$${}_2F_1(a, b; c; z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k z^k}{(c)_k k!} \tag{A.13}$$

for $|z| < 1$ and $\mathrm{Re}(c - a - b) > 0$, where the notation $(x)_n = x(x+1)(x+2)\dots(x+n-1)$ denotes the rising factorial of x.

Appendix B

Gene regulation models

B.1 Birth-death model

Due to the large (possibly infinite) dimensionality of the chemical master equation (2.56), it is not generally solvable in closed form. One notable exception is the so-called "birth-death" model, describing a simple process whereby a molecule of a species X can be either produced (birth) or destroyed (death) with a certain probability:

$$R_1 : \emptyset \xrightarrow{\alpha} X$$
$$R_2 : X \xrightarrow{\beta} \emptyset$$

According to Table 2.1, the corresponding reaction propensities are given by $a_1(\mathbf{X}) = \alpha$, and $a_2(\mathbf{X}) = \beta\, X$, where we assume that the reaction volume is constant and can thus be absorbed into the rate constant α in the case of the zeroth order reaction.

For this system, the CME (4.1) is given by:

$$\frac{\partial}{\partial t}P(X,t) = \alpha P(X-1,t) + \beta\,(X+1)\,P(X+1,t) - (\alpha + \beta\, X)P(X,t) \tag{B.1}$$

This system can be solved analytically by using the moment-generating function G of the system, defined as in (2.39). To compute the solution to the birth-death model (B.1), one can multiply both sides by Z^X, and sum over X to obtain the moment generating function in differential form:

$$\begin{aligned} G(Z,t) &= \sum_{X=0}^{\infty} P(X,t) Z^X \\ \frac{d}{dt}G(Z,t) &= \sum_{X=0}^{\infty} \dot{P}(X,t) Z^X \\ &= \sum_{X=0}^{\infty} \left[\alpha P(X-1,t) + \beta\,(X+1)\,P(X+1,t) - (\alpha + \beta\, X)P(X,t)\right] Z^X \end{aligned} \tag{B.2}$$

Defining $X' = X - 1$ and $X'' = X + 1$, we obtain

$$\sum_{X=0}^{\infty} \dot{P}(X,t)Z^X = \sum_{X'=0}^{\infty} \alpha P(X',t) + \sum_{X''=0}^{\infty} \beta\, X''\, P(X'',t) \tag{B.3}$$

$$- \sum_{X=0}^{\infty} \alpha P(X,t)Z^X - \sum_{X=0}^{\infty} \beta X P(X,t)Z^X \tag{B.4}$$

$$= \alpha G Z + \beta \frac{\partial}{\partial Z} G - \alpha G - \beta Z \frac{\partial}{\partial Z} G \tag{B.5}$$

$$= \left[\alpha(Z-1) + \beta(1-Z)\frac{\partial}{\partial Z}\right] G \tag{B.6}$$

$$= -(1-Z)\left[\alpha - \beta \frac{\partial}{\partial Z}\right] G \tag{B.7}$$

Thus the steady state solution G_∞ of the moment-generating function can be immediately obtained by solving for $\dot{G} = 0$:

$$-(1-Z)\left[\alpha - \beta \frac{\partial}{\partial Z}\right] G_\infty = 0 \tag{B.8}$$

$$\frac{\partial}{\partial Z} G_\infty = \frac{\alpha}{\beta} G_\infty \tag{B.9}$$

for which $G_\infty(Z) = G_0\, \mathrm{e}^{\alpha/\beta Z}$. The normalizing constant G_0 is obtained by requiring that

$$G_\infty(1) = G_0\, \mathrm{e}^{\alpha/\beta} = \sum_{X=0}^{\infty} P(X,t) = 1 \tag{B.10}$$

by which we obtain $G_0 = \mathrm{e}^{-\alpha/\beta}$ and $G_\infty(Z) = e^{\alpha/\beta(Z-1)}$. Finally, the steady state distribution can be obtained using (2.40):

$$P_\infty(X=n) = \frac{\partial^n}{n!\partial Z^n} G_\infty(0) \tag{B.11}$$

$$= \frac{\partial^n}{n!\partial Z^n}\, \mathrm{e}^{\alpha/\beta(Z-1)}\,|_{Z=0} \tag{B.12}$$

$$= \frac{1}{n!}\left(\frac{\alpha}{\beta}\right)^n \mathrm{e}^{-\alpha/\beta} \tag{B.13}$$

Thus the steady state distribution has the form of a Poisson distribution, see (A.8).

The full time-dependent solution can be computed using the method of characteristics, as described in Walczak [230].

B.2 Two-stage model (mRNA and protein)

The birth/death process described in B.1 is useful due to its simplicity. However, it is generally too simple to model the stochastic production of a gene product. In particular, proteins are produced via the translation of mRNAs, which are in turn the results of the

regulated transcription of genes. A *two-stage model* explicitly models mRNA dynamics, while assuming constantly active DNA [111, 237]. A three-stage model allows DNA to transition between active and inactive states while mRNA is only produced when the DNA is active [74, 238].

Consider a system with DNA that is always active and random variables m and n, denoting the number of mRNAs and proteins, respectively, at some time. This system is described by the following chemical reaction network:

$$\begin{aligned} R_1 &: m \xrightarrow{\nu_0} m+1 \\ R_2 &: m \xrightarrow{d_0\, m} m-1 \\ R_3 &: p \xrightarrow{\nu_1\, m} p+1 \\ R_4 &: p \xrightarrow{d_1\, p} p-1 \end{aligned}$$

with constants ν_0, ν_1, d_0, d_1 corresponding to the rate constants for transcription, translation, death of mRNA and death of protein, respectively. The probability density of the system, denoted $P_{m,n}(t)$ corresponding to the number of mRNAs (m) and proteins (n) at time t, evolves according to the following master equation:

$$\begin{aligned} \frac{\partial P_{m,n}}{\partial t} = \nu_0(P_{m-1,n} - P_{m,n}) + \nu_1 m(P_{m,n-1} - P_{m,n}) \\ + d_0[(m+1)P_{m+1,n} - mP_{m,n}] \\ + d_1[(n+1)P_{m,n+1} - nP_{m,n}] \end{aligned} \tag{B.14}$$

Shahrezai *et al.* solve (B.14) approximately for the limit of fast mRNA degradation. They rewrite the CME as a partial differential equation of the (two-variable) generating function (see 2.3.2), defined as $F(z', z) = \sum_{m,n} z'^m z^n P_{m,n}$ for some dummy variables z, z' (corresponding to mRNa and protein, respectively), and solving the resulting PDE in z and z' using the method of characteristics. Assuming fast mRNA degradation, the majority of the probability mass is in the $P_{0,n}$ state, that is, mRNA is mainly entirely absent due to its rapid degradation. Thus, the PDE for the generating function reduces to a function of z alone, with solution

$$F(z, \tau) = \left[\frac{1 - b(z-1)\,\mathrm{e}^{-\tau}}{1 + b - bz}\right]^a \tag{B.15}$$

using rescaled variables $a = \nu_0/d_1, b = \nu_1/d_0, \tau = d_1 t$. Finally, the protein distribution is computed by successive differentiation of the generating function:

$$P_n(\tau) = \frac{1}{n!}\frac{\partial^n}{\partial z^n} F(z, \tau)|_{z=0}. \tag{B.16}$$

Following simplification of the resulting expression, one achieves the approximate, closed-form, time-dependent probability distribution for proteins:

$$P_n(\tau) = \left(\frac{b}{1+b}\right)^n \left(\frac{1 + b\,\mathrm{e}^{-\tau}}{1+b}\right)^a \frac{\Gamma(a+n)}{\Gamma(n)\Gamma(a)} {}_2F_1\left(-n, -a, 1-a-n; \frac{1+b}{\mathrm{e}^{\tau}+b}\right). \tag{B.17}$$

The distribution (B.17) is valid in the regime of fast mRNA degradation ($\gamma = d_0/d_1 \gg 1$) and after relaxation of transient dynamics ($\tau \gg \gamma^{-1}$). For $\tau \gg 1$, (B.17) converges to its steady state given by:

$$P_n = \frac{\Gamma(a+n)}{\Gamma(n)\Gamma(a)} \left(\frac{b}{1+b}\right)^n \left(1 - \frac{b}{1+b}\right)^a. \tag{B.18}$$

The distribution (B.17) can be interpreted intuitively as follows: a given molecule of mRNA can either be degraded or translated within some time interval. Translation occurs with propensity v_1 and the mRNA degradation with propensity d_0, thus the probability of a protein translation event occurring is given by $v_1/(v_1 + d_0) = b/(1 + b)$, see (A.7). The probability of r proteins being translated followed by subsequent degradation of the mRNA molecule is thus given by:

$$P(\Delta P = r) = \left(\frac{b}{1+b}\right)^r \left(1 - \frac{b}{1+b}\right) \tag{B.19}$$

which is the probability of a "burst" of proteins of size r being translated from a single mRNA during the lifetime of the mRNA. The moment generating function of the burst distribution (B.19) is given by the $f(z) = (1 + b - bz)^{-1}$. Since the translation of proteins from distinct mRNA molecules is independent, the generating function for the sum of the proteins over all mRNAs is equal to the product of their respective generating functions. Assuming a mRNA molecules, where $a = \frac{\nu_0}{d_1}$ is the expected number of mRNAs, the generating function for the total number of proteins is given by:

$$\prod_{i=1}^{a} [f(z)]^a = (1 + b - bz)^{-a}. \tag{B.20}$$

The solution to the compound generating function (B.20) is equal to the steady state protein distribution (B.18).

The analytical results summarized here crucially depend on the scale separation parameter γ, between mRNA and protein degradation rates—the simplifying assumptions are only valid in the limit of short mRNA half-lives, which is typically the case for prokaryotes such as budding yeast, but not necessarily for eukaryotic cells. In Section 4, I explore the utility of a method which extends the analytical techniques of [111] to the domain of diminishing scale separation ($\gamma \to 1$).

Bibliography

[1] Martin J Evans and Matthew H Kaufman. Establishment in culture of pluripotential cells from mouse embryos. *Nature*, 292(5819):154–156, 1981.

[2] Roy Williams, Bernhard Schuldt, and Franz-Josef Müller. A guide to stem cell identification: Progress and challenges in system-wide predictive testing with complex biomarkers. *BioEssays*, 33(11):880–890, September 2011.

[3] G R Martin. Isolation of a pluripotent cell line from early mouse embryos cultured in medium conditioned by teratocarcinoma stem cells. *Proceedings of the National Academy of Sciences*, 78(12):7634–7638, December 1981.

[4] R L Williams, D J Hilton, S Pease, T A Willson, C L Stewart, D P Gearing, E F Wagner, D Metcalf, N A Nicola, and N M Gough. Myeloid leukaemia inhibitory factor maintains the developmental potential of embryonic stem cells. *Nature*, 336 (6200):684–687, December 1988.

[5] Austin G Smith. Culture and differentiation of embryonic stem cells. *Journal of tissue culture methods*, 13(2):89–94, 1991.

[6] Qi-Long Ying, Jennifer Nichols, I Chambers, and Austin G Smith. BMP induction of Id proteins suppresses differentiation and sustains embryonic stem cell self-renewal in collaboration with STAT3. *Cell*, 115(3):281–292, October 2003.

[7] Christoffer Tamm, Sara Pijuan Galitó, and Cecilia Annerén. A Comparative Study of Protocols for Mouse Embryonic Stem Cell Culturing. *PLoS One*, 8(12):e81156, December 2013.

[8] Anup Som, Clemens Harder, Boris Greber, Marcin Siatkowski, Yogesh Paudel, Gregor Warsow, Clemens Cap, Hans Schöler, and Georg Fuellen. The PluriNetWork: an electronic representation of the network underlying pluripotency in mouse, and its applications. *PLoS One*, 5(12):e15165, 2010.

[9] I Chambers and S R Tomlinson. The transcriptional foundation of pluripotency. *Development*, 136(14):2311–2322, 2009.

[10] Nicola Festuccia, Rodrigo Osorno, Florian Halbritter, Violetta Karwacki-Neisius, Pablo Navarro, Douglas Colby, Frederick Wong, Adam Yates, Simon R Tomlinson, and I Chambers. Esrrb Is a Direct Nanog Target Gene that Can Substitute for Nanog Function in Pluripotent Cells. *Cell Stem Cell*, 11(4):477–490, October 2012.

[11] Q Zhou, H Chipperfield, D A Melton, and W H Wong. A gene regulatory network in mouse embryonic stem cells. *Proceedings of the National Academy of Sciences*, 104(42):16438–16443, September 2007.

[12] Sridhar Rao and Stuart H Orkin. Unraveling the transcriptional network controlling ES cell pluripotency. *Genome biology*, 7(8):230, 2006.

[13] Kaoru Mitsui, Yoshimi Tokuzawa, Hiroaki Itoh, Kohichi Segawa, Mirei Murakami, Kazutoshi Takahashi, Masayoshi Maruyama, Mitsuyo Maeda, and Shinya Yamanaka. The homeoprotein Nanog is required for maintenance of pluripotency in mouse epiblast and ES cells. *Cell*, 113(5):631–642, May 2003.

[14] Dang Vinh Do, Jun Ueda, Daniel M Messerschmidt, Chanchao Lorthongpanich, Yi Zhou, Bo Feng, Guoji Guo, Peiyu J Lin, Md Zakir Hossain, Wenjun Zhang, Akira Moh, Qiang Wu, Paul Robson, Huck-Hui Ng, Lorenz Poellinger, Barbara B Knowles, Davor Solter, and Xin-Yuan Fu. A genetic and developmental pathway from STAT3 to the OCT4-NANOG circuit is essential for maintenance of ICM lineages in vivo. *Genes & development*, 27(12):1378–1390, June 2013.

[15] Angie Rizzino. Concise review: The Sox2-Oct4 connection: critical players in a much larger interdependent network integrated at multiple levels. *Stem Cells*, 31(6): 1033–1039, June 2013.

[16] Qi-Long Ying, Jason Wray, Jennifer Nichols, Laura Batlle-Morera, Bradley Doble, James Woodgett, Philip Cohen, and Austin G Smith. The ground state of embryonic stem cell self-renewal. *Nature*, 453(7194):519–523, May 2008.

[17] LA A Boyer, Tong Ihn Lee, Megan F Cole, Sarah E Johnstone, Stuart S Levine, Jacob P Zucker, Matthew G Guenther, Roshan M Kumar, Heather L Murray, Richard G Jenner, David K Gifford, Douglas A Melton, Rudolf Jaenisch, and Richard A Young. Core transcriptional regulatory circuitry in human embryonic stem cells. *Cell*, 122(6):947–956, September 2005.

[18] Jianlong Wang, Sridhar Rao, Jianlin Chu, Xiaohua Shen, Dana N Levasseur, Thorold W Theunissen, and Stuart H Orkin. A protein interaction network for pluripotency of embryonic stem cells. *Nature*, 444(7117):364–368, November 2006.

[19] Arven Saunders, Francesco Faiola, and Jianlong Wang. Concise Review: Pursuing Self-Renewal and Pluripotency with the Stem Cell Factor Nanog. *Stem Cells*, 31(7): 1227–1236, July 2013.

[20] Jonghwan Kim, Jianlin Chu, Xiaohua Shen, Jianlong Wang, and Stuart H Orkin. An Extended Transcriptional Network for Pluripotency of Embryonic Stem Cells. *Cell*, 132(6):1049–1061, March 2008.

[21] Rudolf Jaenisch and Richard Young. Stem Cells, the Molecular Circuitry of Pluripotency and Nuclear Reprogramming. *Cell*, 132(4):567–582, February 2008.

[22] Miguel Fidalgo, Francesco Faiola, Carlos-Filipe Pereira, Junjun Ding, Arven Saunders, Julian Gingold, Christoph Schaniel, Ihor R Lemischka, José C R Silva, and Jianlong Wang. Zfp281 mediates Nanog autorepression through recruitment of the NuRD complex and inhibits somatic cell reprogramming. *Proceedings of the National Academy of Sciences*, 109(40):16202–16207, October 2012.

[23] Pablo Navarro, Nicola Festuccia, Douglas Colby, Alessia Gagliardi, Nicholas P Mullin, Wensheng Zhang, Violetta Karwacki-Neisius, Rodrigo Osorno, David Kelly, Morag Robertson, and I Chambers. OCT4/SOX2-independent Nanog autorepression modulates heterogeneous Nanog gene expression in mouse ES cells. *The EMBO Journal*, November 2012.

[24] I Chambers, José C R Silva, Douglas Colby, Jennifer Nichols, Bianca Nijmeijer, Morag Robertson, Jan Vrana, Ken Jones, Lars Grotewold, and Austin G Smith. Nanog safeguards pluripotency and mediates germline development. *Nature*, 450 (7173):1230–1234, December 2007.

[25] T Kalmar, C Lim, P Hayward, and Silvia Muñoz-Descalzo. Regulated fluctuations in nanog expression mediate cell fate decisions in embryonic stem cells. *PLoS Biology*, 2009.

[26] Elsa Abranches, A M V Guedes, M Moravec, H Maamar, P Svoboda, Arjun Raj, and D Henrique. Stochastic NANOG fluctuations allow mouse embryonic stem cells to explore pluripotency. *Development*, 141(14):2770–2779, July 2014.

[27] Amar M Singh, Takashi Hamazaki, Katherine E Hankowski, and Naohiro Terada. A Heterogeneous Expression Pattern for Nanog in Embryonic Stem Cells. *Stem Cells*, 25(10):2534–2542, October 2007.

[28] Sui Huang. Non-genetic heterogeneity of cells in development: more than just noise. *Development*, 136(23):3853–3862, December 2009.

[29] Ingmar Glauche, Maria Herberg, and Ingo Roeder. Nanog variability and pluripotency regulation of embryonic stem cells-insights from a mathematical model analysis. *PLoS One*, 5(6):e11238, 2010.

[30] Chikara Furusawa and Kunihiko Kaneko. Chaotic expression dynamics implies pluripotency: when theory and experiment meet. *Biology direct*, 4(1):17, 2009.

[31] Zakary S Singer, John Yong, Julia Tischler, Jamie A Hackett, Alphan Altinok, M Azim Surani, Long Cai, and Michael B Elowitz. Dynamic heterogeneity and DNA methylation in embryonic stem cells. *Molecular cell*, 55(2):319–331, July 2014.

[32] Katsuhiko Hayashi, Susana M Chuva de Sousa Lopes, Fuchou Tang, and M Azim Surani. Dynamic Equilibrium and Heterogeneity of Mouse Pluripotent Stem Cells with Distinct Functional and Epigenetic States. *Cell Stem Cell*, 3(4):391–401, October 2008.

[33] Mo Li, Guang-Hui Liu, and Juan Carlos Izpisua Belmonte. Navigating the epigenetic landscape of pluripotent stem cells. *Nature reviews. Molecular cell biology*, 13(8): 524–535, 2012.

[34] Dina A Faddah, Haoyi Wang, Albert Wu Cheng, Yarden Katz, Yosef Buganim, and Rudolf Jaenisch. Single-Cell Analysis Reveals that Expression of Nanog Is Biallelic and Equally Variable as that of Other Pluripotency Factors in Mouse ESCs. *Cell Stem Cell*, 13(1):23–29, July 2013.

[35] Austin G Smith. Nanog Heterogeneity: Tilting at Windmills? *Cell Stem Cell*, 13 (1):6–7, July 2013.

[36] J Silva, O Barrandon, Jennifer Nichols, and J Kawaguchi. Promotion of reprogramming to ground state pluripotency by signal inhibition. *PLoS Biology*, 2008.

[37] Maurice A Canham, Alexei A Sharov, Minoru S H Ko, and Joshua M Brickman. Functional Heterogeneity of Embryonic Stem Cells Revealed through Translational Amplification of an Early Endodermal Transcript. *PLoS Biology*, 8(5):e1000379, May 2010.

[38] Yayoi Toyooka, Daisuke Shimosato, Kazuhiro Murakami, Kadue Takahashi, and Hitoshi Niwa. Identification and characterization of subpopulations in undifferentiated ES cell culture. *Development*, 135(5):909–918, March 2008.

[39] Alfonso Martinez-Arias and Joshua M Brickman. Gene expression heterogeneities in embryonic stem cell populations: origin and function. *Current opinion in cell biology*, 23(6):650–656, December 2011.

[40] Jamie Trott, Katsuhiko Hayashi, Azim Surani, M Madan Babu, and Alfonso Martinez-Arias. Dissecting ensemble networks in ES cell populations reveals microheterogeneity underlying pluripotency. *Molecular BioSystems*, 8(3):744–752, March 2012.

[41] Thomas Graf and Matthias Stadtfeld. Heterogeneity of Embryonic and Adult Stem Cells. *Cell Stem Cell*, 3(5):480–483, November 2008.

[42] Mads Kaern, Timothy C Elston, William J Blake, and James J Collins. Stochasticity in gene expression: from theories to phenotypes. *Nature Reviews Genetics*, 6(6):451–464, June 2005.

[43] C. H. Waddington. *The strategy of the genes. A discussion of some aspects of theoretical biology. With an appendix by H. Kacser.* London: George Allen & Unwin, Ltd., 1957.

[44] Sui Huang. The molecular and mathematical basis of Waddington's epigenetic landscape: A framework for post-Darwinian biology? *BioEssays*, 34(2):149–157, November 2011.

[45] Cheng Lv, Xiaoguang Li, Fangting Li, and Tiejun Li. Constructing the Energy Landscape for Genetic Switching System Driven by Intrinsic Noise. *PLoS One*, 9 (2):e88167, February 2014.

[46] Masaki Sasai, Yudai Kawabata, Koh Makishi, Kazuhito Itoh, and Tomoki P Terada. Time Scales in Epigenetic Dynamics and Phenotypic Heterogeneity of Embryonic Stem Cells. *PLoS Computational Biology*, 9(12):e1003380, December 2013.

[47] Christopher RS Banerji, Diego Miranda-Saavedra, Simone Severini, Martin Widschwendter, Tariq Enver, Joseph Xu Zhou, and Andrew E Teschendorff. Cellular network entropy as the energy potential in Waddington's differentiation landscape. *Scientific reports*, 3, 2013.

[48] Ben D MacArthur and Ihor R Lemischka. Statistical Mechanics of Pluripotency. *Cell*, 154(3):484–489, August 2013.

[49] Chunhe Li and Jin Wang. Quantifying Cell Fate Decisions for Differentiation and Reprogramming of a Human Stem Cell Network: Landscape and Biological Paths. *PLoS Computational Biology*, 9(8):e1003165, August 2013.

[50] Joseph Xu Zhou, M D S Aliyu, Erik Aurell, and Sui Huang. Quasi-potential landscape in complex multi-stable systems. *Journal of the Royal Society, Interface / the Royal Society*, 9(77):3539–3553, December 2012.

[51] Jincheng Wu and Emmanuel S Tzanakakis. Deconstructing stem cell population heterogeneity: single-cell analysis and modeling approaches. *Biotechnology advances*, 31(7):1047–1062, November 2013.

[52] Carolyn M Southward and Michael G Surette. The dynamic microbe: green fluorescent protein brings bacteria to light. *Molecular microbiology*, 45(5):1191–1196, September 2002.

[53] Jin Zhang, Robert E Campbell, Alice Y Ting, and Roger Y Tsien. Creating new fluorescent probes for cell biology. *Nature reviews. Molecular cell biology*, 3(12): 906–918, December 2002.

[54] Enrico Lugli, Mario Roederer, and Andrea Cossarizza. Data analysis in flow cytometry: The future just started. *Cytometry*, 77A(7):705–713, April 2010.

[55] M G Ormerod. *Flow Cytometry.* A Practical Approach. Oxford University Press, May 2000.

[56] Dmitry R Bandura, Vladimir I Baranov, Olga I Ornatsky, Alexei Antonov, Robert Kinach, Xudong Lou, Serguei Pavlov, Sergey Vorobiev, John E Dick, and Scott D Tanner. Mass cytometry: technique for real time single cell multitarget immunoassay based on inductively coupled plasma time-of-flight mass spectrometry. *Analytical chemistry*, 81(16):6813–6822, August 2009.

[57] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

[58] El-ad David Amir, Kara L Davis, Michelle D Tadmor, Erin F Simonds, Jacob H Levine, Sean C Bendall, Daniel K Shenfeld, Smita Krishnaswamy, Garry P Nolan, and Dana Pe'er. viSNE enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. *Nature biotechnology*, 31(6):545–552, June 2013.

[59] Peng Qiu, Erin F Simonds, Sean C Bendall, Kenneth D Gibbs, Robert V Bruggner, Michael D Linderman, Karen Sachs, Garry P Nolan, and Sylvia K Plevritis. Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. *Nature biotechnology*, 29(10):886–891, October 2011.

[60] Florian Buettner and Fabian J Theis. A novel approach for resolving differences in single-cell gene expression patterns from zygote to blastocyst. *Bioinformatics*, 28 (18):i626–i632, September 2012.

[61] Laleh Haghverdi, Florian Buettner, and Fabian J Theis. Diffusion maps for high-dimensional single-cell analysis of differentiation data. *Bioinformatics*, 31(18):2989–2998, May 2015.

[62] Michael Schwarzfischer, Oliver Hilsenbeck, Bernhard Schauberger, Sabine Hug, Adam Filipczyk, Philipp Hoppe, Michael Strasser, Felix Buggenthin, Justin Feigelman, Jan Krumsiek, Dirk Löffler, Konstantinos Kokkaliaris, Adrianus J J van den Berg, Max Endele, J Hasenauer, Carsten Marr, F J Theis, and Timm Schroeder. Reliable long-term single-cell tracking and quantification of cellular and molecular behavior in time-lapse microscopy .

[63] Diane M Longo and Jeff Hasty. Dynamics of single-cell gene expression. *Molecular systems biology*, 2(1):64, 2006.

[64] X Michalet, F F Pinaud, L A Bentolila, J M Tsay, S Doose, J J Li, G Sundaresan, A M Wu, S S Gambhir, and S Weiss. Quantum dots for live cells, in vivo imaging, and diagnostics. *Science*, 307(5709):538–544, January 2005.

[65] Iftach Nachman, Aviv Regev, and Sharad Ramanathan. Dissecting Timing Variability in Yeast Meiosis. *Cell*, 131(3):544–556, November 2007.

[66] Stefano Di Talia, Jan M Skotheim, James M Bean, Eric D Siggia, and Frederick R Cross. The effects of molecular noise and size control on variability in the budding yeast cell cycle. *Nature*, 448(7156):947–951, August 2007.

[67] Jan M Skotheim, Stefano Di Talia, Eric D Siggia, and Frederick R Cross. Positive feedback of G1 cyclins ensures coherent cell cycle entry. *Nature*, 454(7202):291–296, July 2008.

[68] E Dultz, E Zanin, C Wurzenberger, M Braun, G Rabut, L Sironi, and J Ellenberg. Systematic kinetic analysis of mitotic dis- and reassembly of the nuclear pore in living cells. *The Journal of cell biology*, 180(5):857–865, March 2008.

[69] Brian Munsky, Brooke Trinh, and Mustafa Khammash. Listening to the noise: random fluctuations reveal gene network parameters. *Molecular systems biology*, 5, October 2009.

[70] Mary J Dunlop, Robert Sidney Cox, Joseph H Levine, Richard M Murray, and Michael B Elowitz. Regulatory activity revealed by dynamic correlations in gene expression noise. *Nature genetics*, 40(12):1493–1498, November 2008.

[71] Dale Muzzey and Alexander van Oudenaarden. Quantitative Time-Lapse Fluorescence Microscopy in Single Cells. *Annual Review of Cell and Developmental Biology*, 25(1):301–327, November 2009.

[72] Kiyomi Taniguchi, Tomoharu Kajiyama, and Hideki Kambara. Quantitative analysis of gene expression in a single cell by qPCR. *Nature methods*, 6(7):503–506, July 2009.

[73] R R Swiger and J D Tucker. Fluorescence in situ hybridization: a brief review. *Environmental and molecular mutagenesis*, 1996.

[74] Arjun Raj, Charles S Peskin, Daniel Tranchina, Diana Y Vargas, and Sanjay Tyagi. Stochastic mRNA Synthesis in Mammalian Cells. *PLoS Biology*, 4(10):e309, September 2006.

[75] Rickard Sandberg. Entering the era of single-cell transcriptomics in biology and medicine. *Nature methods*, 11(1):22–24, January 2014.

[76] Aaron M Streets, Xiannian Zhang, Chen Cao, Yuhong Pang, Xinglong Wu, Liang Xiong, Lu Yang, Yusi Fu, Liang Zhao, Fuchou Tang, and Yanyi Huang. Microfluidic single-cell whole-transcriptome sequencing. *Proceedings of the National Academy of Sciences*, 111(19):7048–7053, 2014.

[77] Allon M Klein, Linas Mazutis, Ilke Akartuna, Naren Tallapragada, Adrian Veres, Victor Li, Leonid Peshkin, David A Weitz, and Marc W Kirschner. Droplet Barcoding for Single-Cell Transcriptomics Applied to Embryonic Stem Cells. *Cell*, 161 (5):1187–1201, May 2015.

[78] Evan Z Macosko, Anindita Basu, Rahul Satija, James Nemesh, Karthik Shekhar, Melissa Goldman, Itay Tirosh, Allison R Bialas, Nolan Kamitaki, Emily M Martersteck, John J Trombetta, David A Weitz, Joshua R Sanes, Alex K Shalek, Aviv Regev, and Steven A McCarroll. Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets. *Cell*, 161(5):1202–1214, May 2015.

[79] S M Janicki, T Tsukamoto, S E Salghetti, W P Tansey, R Sachidanandam, K V Prasanth, T Ried, Y Shav-Tal, E Bertrand, R H Singer, and D L Spector. From silencing to gene expression: Real-time analysis in single cells. *Cell*, 116(5):683–698, 2004.

[80] Daojing Wang and Steven Bodovitz. Single cell analysis: the new frontier in 'omics'. *Trends in biotechnology*, 28(6):281–290, June 2010.

[81] Fuchou Tang, Kaiqin Lao, and M Azim Surani. Development and applications of single-cell transcriptome analysis. *Nature methods*, 8(4 Suppl):S6–11, April 2011.

[82] Westbrook M Weaver, Peter Tseng, Anja Kunze, Mahdokht Masaeli, Aram J Chung, Jaideep S Dudani, Harsha Kittur, Rajan P Kulkarni, and Dino Di Carlo. Advances in high-throughput single-cell microtechnologies. *Current opinion in biotechnology*, 25:114–123, February 2014.

[83] Tomer Kalisky, Paul Blainey, and Stephen R Quake. Genomic Analysis at the Single-Cell Level. *Annual Review of Genetics*, 45(1):431–445, December 2011.

[84] Arjun Raj and Alexander van Oudenaarden. Nature, nurture, or chance: stochastic gene expression and its consequences. *Cell*, 135(2):216–226, October 2008.

[85] Michael B Elowitz, Arnold J Levine, Eric D Siggia, and Peter S Swain. Stochastic gene expression in a single cell. *Science*, 297(5584):1183–1186, August 2002.

[86] Andreas Hilfinger and Johan Paulsson. Separating intrinsic from extrinsic fluctuations in dynamic biological systems. *Proceedings of the National Academy of Sciences*, 108(29):12167–12172, July 2011.

[87] Gábor Balázsi, Alexander van Oudenaarden, and James J Collins. Cellular decision making and biological noise: from microbes to mammals. *Cell*, 144(6):910–925, March 2011.

[88] Avigdor Eldar and Michael B Elowitz. Functional roles for noise in genetic circuits. *Nature*, 467(7312):167–173, September 2010.

[89] Robert A Beckman, Gunter S Schemmann, and Chen-Hsiang Yeang. Impact of genetic dynamics and single-cell heterogeneity on development of nonstandard personalized medicine strategies for cancer. *Proceedings of the National Academy of Sciences*, 109(36):14586–14591, September 2012.

[90] Mei-Chong Wendy Lee, Fernando J Lopez-Diaz, Shahid Yar Khan, Muhammad Akram Tariq, Yelena Dayn, Charles Joseph Vaske, Amie J Radenbaugh, Hyunsung John Kim, Beverly M Emerson, and Nader Pourmand. Single-cell analyses of transcriptional heterogeneity during drug tolerance transition in cancer cells by RNA sequencing. *Proceedings of the National Academy of Sciences*, October 2014.

[91] Berend Snijder and Lucas Pelkmans. Origins of regulated cell-to-cell variability. *Nature reviews. Molecular cell biology*, 12(2):119–125, January 2011.

[92] Darren J Wilkinson. Stochastic modelling for quantitative description of heterogeneous biological systems. *Nature Reviews Genetics*, 10(2):122–133, February 2009.

[93] Jerome T Mettetal. Predicting stochastic gene expression dynamics in single cells. *Proceedings of the National Academy of Sciences*, 103(19):7304–7309, April 2006.

[94] Rajesh Ramaswamy, Nélido González-Segredo, Ivo F Sbalzarini, and Ramon Grima. Discreteness-induced concentration inversion in mesoscopic chemical systems. *Nature communications*, 3:779, 2012.

[95] David Angeli. A Tutorial on Chemical Reaction Network Dynamics. *European journal of control*, 15(3-4):398–406, January 2009.

[96] Daniel T Gillespie. A rigorous derivation of the chemical master equation. *Physica A: Statistical Mechanics and its Applications*, 188(1-3):404–425, September 1992.

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

[98] Stefan Engblom. Computing the moments of high dimensional solutions of the master equation. *Applied Mathematics and Computation*, 180(2):498–515, September 2006.

[99] Srividya I Iyer-Biswas, F Hayot, and C Jayaprakash. Stochasticity of gene products from transcriptional pulsing. *Physical Review E*, 79(3 Pt 1):031911, March 2009.

[100] Srividya I Iyer-Biswas and C Jayaprakash. Mixed Poisson distributions in exact solutions of stochastic autoregulation models. *Physical Review E*, 90(5):052712, November 2014.

[101] Tomás Aquino, Elsa Abranches, and Ana Nunes. Stochastic single-gene autoregulation. *Physical Review E*, 85(6):061913, June 2012.

[102] Vlad Elgart, Tao Jia, Andrew T Fenley, and Rahul Kulkarni. Connecting protein and mRNA burst distributions for stochastic models of gene expression. *Physical biology*, 8(4):046001, April 2011.

[103] Yves Vandecan and Ralf Blossey. Self-regulatory gene: an exact solution for the gene gate model. *Physical Review E*, 87(4):042705, April 2013.

[104] Ramon Grima, D R Schmidt, and T J Newman. Steady-state fluctuations of a genetic feedback loop: an exact solution. *The Journal of chemical physics*, 137(3): 035104, July 2012.

[105] Mukund Thattai and Alexander van Oudenaarden. Intrinsic noise in gene regulatory networks. *Proceedings of the National Academy of Sciences*, 98(15):8614–8619, 2001.

[106] Nir Friedman, Long Cai, and X Xie. Linking Stochastic Dynamics to Population Distribution: An Analytical Framework of Gene Expression. *Physical review letters*, 97(16):168302, October 2006.

[107] Bence Mélykúti, J P Hespanha, and Mustafa Khammash. Equilibrium distributions of simple biochemical reaction systems for time-scale separation in stochastic reaction networks. *Journal of the Royal Society, Interface / the Royal Society*, 11(97): 20140054–20140054, May 2014.

[108] A F Ramos, G C P Innocentini, and J E M Hornos. Exact time-dependent solutions for a self-regulating gene. *Physical Review E*, 83(6):062902, June 2011.

[109] Johan Paulsson. Models of stochastic gene expression. *Physics of Life Reviews*, 2 (2):157–175, June 2005.

[110] Aleksandra M. M Walczak, Andrew Mugler, and Chris H. WIggins. Analytic methods for modeling stochastic regulatory networks. *Methods in molecular biology (Clifton, N.J.)*, 880:273–322, 2012.

[111] Vahid Shahrezaei and Peter S Swain. Analytical distributions for stochastic gene expression. *Proceedings of the National Academy of Sciences*, 105(45):17256–17261, November 2008.

[112] Nikola Popović, Carsten Marr, and Peter S Swain. A geometric analysis of fast-slow models for stochastic gene expression. *Journal of Mathematical Biology*, pages 1–36, April 2015.

[113] Pavol Bokes, John R King, Andrew T A Wood, and Matthew Loose. Exact and approximate distributions of protein and mRNA levels in the low-copy regime of gene expression. *Journal of Mathematical Biology*, 64(5):829–854, June 2011.

[114] D Huh and Johan Paulsson. Non-genetic heterogeneity from stochastic partitioning at cell division. *Nature genetics*, 2010.

[115] Dann Huh and Johan Paulsson. Random partitioning of molecules at cell division. *Proceedings of the National Academy of Sciences*, 108(36):15004–15009, September 2011.

[116] Nikos V Mantzaris. From single-cell genetic architecture to cell population dynamics: quantitatively decomposing the effects of different population heterogeneity sources for a genetic network with positive feedback architecture. *Biophysical journal*, 92 (12):4271–4288, June 2007.

[117] Daniel T Gillespie. Exact stochastic simulation of coupled chemical reactions . *The journal of physical chemistry*, 81(25), 1977.

[118] Yang Cao, Hong Li, and Linda Petzold. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *The Journal of chemical physics*, 121(9):4059–4067, 2004.

[119] Rajesh Ramaswamy, Nélido González-Segredo, and Ivo F Sbalzarini. A new class of highly efficient exact stochastic simulation algorithms for chemical reaction networks. *The Journal of chemical physics*, 130(24):244104, June 2009.

[120] Anne Auger, Philippe Chatelain, and Petros Koumoutsakos. R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *The Journal of chemical physics*, 125(8):084103, 2006.

[121] Xin-jun Peng and Yi-fei Wang. L-leap: accelerating the stochastic simulation of chemically reacting systems. *Applied Mathematics and Mechanics*, 28(10):1361–1371, October 2007.

[122] Daniel T Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *The Journal of chemical physics*, 115(4):1716–1733, 2001.

[123] Muruhan Rathinam, Linda R Petzold, Yang Cao, and Daniel T Gillespie. Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method. *The Journal of chemical physics*, 119(24):12784–12794, December 2003.

[124] Xiaodong Cai and Zhouyi Xu. K-leap method for accelerating stochastic simulation of coupled chemical reactions. *The Journal of chemical physics*, 126(7):074102, 2007.

[125] Daniel T Gillespie. The chemical Langevin equation. *The Journal of chemical physics*, 2000.

[126] Thomas Henzinger, Linar Mikeev, Maria Mateescu, and Verena Wolf. Hybrid Numerical Solution of the Chemical Master Equation. In *Computational Methods in Systems Biology*, pages 55–65, September 2010.

[127] Yang Cao, Daniel T Gillespie, and Linda R Petzold. The slow-scale stochastic simulation algorithm. *The Journal of chemical physics*, 122(1):14116, January 2005.

[128] Eric L Haseltine and James B Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *The Journal of chemical physics*, 117(15):6959, 2002.

[129] P Koumoutsakos and Justin Feigelman. Multiscale stochastic simulations of chemical reactions with regulated scale separation. *Journal of Computational Physics*, 2013.

[130] J Pahle. Biochemical simulations: stochastic, approximate stochastic and hybrid approaches. *Briefings in Bioinformatics*, 10(1):53–64, October 2008.

[131] Brian Munsky and Mustafa Khammash. The finite state projection algorithm for the solution of the chemical master equation. *The Journal of chemical physics*, 124 (4):044104–044104–13, 2006.

[132] Vladimir Kazeev, Mustafa Khammash, Michael Nip, and Christoph Schwab. Direct solution of the Chemical Master Equation using quantized tensor trains. *PLoS Computational Biology*, 10(3):e1003359, March 2014.

[133] Verena Wolf, Rushil Goel, Maria Mateescu, and Thomas A Henzinger. Solving the chemical master equation using sliding windows. *BMC systems biology*, 4(1):42, 2010.

[134] Markus Hegland, Conrad Burden, Lucia Santoso, Shev MacNamara, and Hilary Booth. A solver for the stochastic master equation applied to gene regulatory networks. *Journal of Computational and Applied Mathematics*, 205(2):708–724, August 2007.

[135] Andreas Raue, C Kreutz, T Maiwald, J Bachmann, M Schilling, U Klingmüller, and J Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, August 2009.

[136] D S Sivia. *Data Analysis*. A Bayesian Tutorial. Oxford University Press, 1996.

[137] Andrew Golightly and Darren J Wilkinson. Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte Carlo. *Interface focus*, 1(6):807–820, December 2011.

[138] Christoph Zechner, S Pelet, M Peter, and H Koeppl. Recursive Bayesian estimation of stochastic rate constants from heterogeneous cell populations. In *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC 2011)*, pages 5837–5843. IEEE, 2011.

[139] R J Boys, Darren J Wilkinson, and T B L Kirkwood. Bayesian inference for a discretely observed stochastic kinetic model. *Statistics and Computing*, 18(2):125–135, November 2007.

[140] Michael Amrein and Hans R Künsch. Rate estimation in partially observed Markov jump processes with measurement errors. *Statistics and Computing*, 22(2):513–526, March 2011.

[141] A Golightly and Darren J Wilkinson. Bayesian inference for stochastic kinetic models using a diffusion approximation. *Biometrics*, 61(3):781–788, September 2005.

[142] Angelique Ale, Paul Kirk, and Michael P H Stumpf. A general moment expansion method for stochastic kinetic models. *The Journal of chemical physics*, 138(17): 174101, May 2013.

[143] Chang Hyeong Lee, Kyeong-Hun Kim, and Pilwon Kim. A moment closure method for stochastic reaction networks. *The Journal of chemical physics*, 130(13):134107, April 2009.

[144] Joao Hespanha. Moment closure for biochemical networks. In *International Symposium on Communications, Control and Signal Processing*, pages 142–147, November 2008.

[145] Ramon Grima. A study of the accuracy of moment-closure approximations for stochastic chemical kinetics. *The Journal of chemical physics*, 136(15):154105, April 2012.

[146] Vijay Chickarmane and Carsten Peterson. A computational model for understanding stem cell, trophectoderm and endoderm lineage determination. *PLoS One*, 3(10): e3478, 2008.

[147] Hiroshi Ochiai, Takeshi Sugawara, Tetsushi Sakuma, and Takashi Yamamoto. Stochastic promoter activation affects Nanog expression variability in mouse embryonic stem cells. *Scientific reports*, 4:7125, 2014.

[148] Maria Herberg, Tuzer Kalkan, Ingmar Glauche, Austin G Smith, and Ingo Roeder. A Model-Based Analysis of Culture-Dependent Phenotypes of mESCs. *PLoS One*, 9(3):e92496, March 2014.

[149] Vijay Chickarmane, Carl Troein, Ulrike A Nuber, Herbert M Sauro, and Carsten Peterson. Transcriptional dynamics of the embryonic stem cell switch. *PLoS Computational Biology*, 2(9):e123, September 2006.

[150] Adam Filipczyk, Carsten Marr, Simon Hastreiter, Justin Feigelman, Michael Schwarzfischer, Philipp S Hoppe, Dirk Loeffler, Konstantinos D Kokkaliaris, Max Endele, Bernhard Schauberger, Oliver Hilsenbeck, Stavroula Skylaki, Jan Hasenauer, Konstantinos Anastassiadis, Fabian J Theis, and Timm Schroeder. Network plasticity of pluripotency transcription factors in embryonic stem cells. *Nature cell biology*, September 2015.

[151] Kathryn N Ivey, Alecia Muth, Joshua Arnold, Frank W King, Ru-Fang Yeh, Jason E Fish, Edward C Hsiao, Robert J Schwartz, Bruce R Conklin, Harold S Bernstein, and Deepak Srivastava. MicroRNA Regulation of Cell Lineages in Mouse and Human Embryonic Stem Cells. *Cell Stem Cell*, 2(3):219–229, March 2008.

[152] Jincheng Wu and Emmanuel S Tzanakakis. Distinct Allelic Patterns of Nanog Expression Impart Embryonic Stem Cell Population Heterogeneity. *PLoS Computational Biology*, 9(7):e1003140, July 2013.

[153] Yusuke Miyanari and Maria-Elena Torres-Padilla. Control of ground-state pluripotency by allelic regulation of Nanog. *Nature*, 483(7390):470–473, March 2012.

[154] Adam Filipczyk, Konstantinos Gkatzis, Jun Fu, Philipp S Hoppe, Heiko Lickert, Konstantinos Anastassiadis, and Timm Schroeder. Biallelic Expression of Nanog Protein in Mouse Embryonic Stem Cells. *Stem Cell*, 13(1):12–13, July 2013.

[155] Bin Zhang and Peter G Wolynes. Stem cell differentiation as a many-body problem. *Proceedings of the National Academy of Sciences*, 111(28):10185–10190, July 2014.

[156] Maria Herberg and Ingo Roeder. Computational modelling of embryonic stem-cell fate control. *Development*, 142(13):2250–2260, July 2015.

[157] Jean Jacod and Philip Protter. *Probability Essentials*. Universitext. Springer Science & Business Media, Berlin, Heidelberg, December 2012.

[158] M G Kendall. *Rank correlation methods.* Griffin, Oxford, England, 1948.

[159] M G Kendall and A Stuart. The Advanced Theory of Statistics. Vol.2: Inference and, 1973.

[160] Michael R Chernick. *Bootstrap Methods*. A Guide for Practitioners and Researchers. John Wiley & Sons, September 2011.

[161] David J C Mackay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, September 2003.

[162] Simon Jackman. Estimation and Inference via Bayesian Simulation: An Introduction to Markov Chain Monte Carlo. *American Journal of Political Science*, 44(2):375–404, June 2007.

[163] Stephen P Brooks and Gareth O Roberts. Convergence assessment techniques for Markov chain Monte Carlo. *Statistics and Computing*, 8:1–17, March 1999.

[164] Andrea Saltelli, Marco Ratto, Terry Andres, Francesca Campolongo, Jessica Cariboni, Debora Gatelli, Michaela Saisana, and Stefano Tarantola. *Introduction to Sensitivity Analysis*. John Wiley & Sons, Ltd, Chichester, UK, 2008.

[165] Andreas Raue, Marcel Schilling, Julie Bachmann, Andrew Matteson, Max Schelker, Max Schelke, Daniel Kaschek, Sabine Hug, Clemens Kreutz, Brian D Harms, Fabian J Theis, Ursula Klingmüller, and Jens Timmer. Lessons learned from quantitative dynamical modeling in systems biology. *PLoS One*, 8(9):e74335, 2013.

[166] Brian Munsky and Mustafa Khammash. The finite state projection algorithm for the solution of the chemical master equation. *The Journal of chemical physics*, 124 (4):044104, January 2006.

[167] N G Van Kampen. *Stochastic Processes in Physics and Chemistry*. Elsevier, Amsterdam, August 2011.

[168] R F Pawula. Approximation of the Linear Boltzmann Equation by the Fokker-Planck Equation. *Physical Review*, 162(1):186–188, October 1967.

[169] Crispin Gardiner. *Stochastic Methods*. A Handbook for the Natural and Social Sciences. Springer, January 2009.

[170] Ramon Grima. An effective rate equation approach to reaction kinetics in small volumes: theory and application to biochemical reactions in nonequilibrium steady-state conditions. *The Journal of chemical physics*, 133(3):035101, July 2010.

[171] Michael A Gibson and Jehoshua Bruck. Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels. *The Journal of Physical Chemistry A*, 104(9):1876–1889, March 2000.

[172] Benjamin Hepp, Ankit Gupta, and Mustafa Khammash. Adaptive hybrid simulations for multiscale stochastic reaction networks. *The Journal of chemical physics*, 142(3):034118, January 2015.

[173] Hiroyuki Kuwahara and Ivan Mura. An efficient and exact stochastic simulation method to analyze rare events in biochemical systems. *The Journal of chemical physics*, 129(16):165101, 2008.

[174] Eric Mjolsness, David Orendorff, Philippe Chatelain, and Petros Koumoutsakos. An exact accelerated stochastic simulation algorithm. *The Journal of chemical physics*, 130(14):144110, 2009.

[175] Kazim H Narsinh, Ning Sun, Veronica Sanchez-Freire, Andrew S Lee, Patricia Almeida, Shijun Hu, Taha Jan, Kitchener D Wilson, Denise Leong, Jarrett Rosenberg, Mylene Yao, Robert C Robbins, and Joseph C Wu. Single cell transcriptional profiling reveals heterogeneity of human induced pluripotent stem cells. *The Journal of clinical investigation*, 121(3):1217–1221, March 2011.

[176] Julia Tischler and M Azim Surani. Investigating transcriptional states at single-cell-resolution. *Current opinion in biotechnology*, 24(1):69–78, October 2012.

[177] Stanislav S Rubakhin, Eric J Lanni, and Jonathan V Sweedler. Progress toward single cell metabolomics. *Current opinion in biotechnology*, 24(1):95–104, February 2013.

[178] Karsten Suhre, So-Youn Shin, Ann-Kristin Petersen, Robert P Mohney, David Meredith, Brigitte Wägele, Elisabeth Altmaier, CARDIoGRAM, Panos Deloukas, Jeanette Erdmann, Elin Grundberg, Christopher J Hammond, Martin Hrabé de Angelis, Gabi Kastenmüller, Anna Köttgen, Florian Kronenberg, Massimo Mangino, Christa Meisinger, Thomas Meitinger, Hans-Werner Mewes, Michael V Milburn, Cornelia Prehn, Johannes Raffler, Janina S Ried, Werner Römisch-Margl, Nilesh J Samani, Kerrin S Small, H Erich Wichmann, Guangju Zhai, Thomas Illig, Tim D Spector, Jerzy Adamski, Nicole Soranzo, and Christian Gieger. Human metabolic individuality in biomedical and pharmaceutical research. *Nature*, 477(7362):54–60, September 2011.

[179] Allison Mayle, Min Luo, Mira Jeong, and Margaret A Goodell. Flow cytometry analysis of murine hematopoietic stem cells. *Cytometry*, 83A(1):27–37, June 2012.

[180] Hannah H Chang, Martin Hemberg, Mauricio Barahona, Donald E Ingber, and Sui Huang. Transcriptome-wide noise controls lineage choice in mammalian progenitor cells. *Nature*, 453(7194):544–547, May 2008.

[181] Robert F Murphy. Automated identification of subpopulations in flow cytometric list mode data using cluster analysis. *Cytometry*, 6(4):302–309, July 1985.

[182] Ali Bashashati and Ryan R Brinkman. A Survey of Flow Cytometry Data Analysis Methods. *Advances in bioinformatics*, 2009(1):1–19, 2009.

[183] Sanjay Goil, Harsha Nagesh, and Alok Choudhary. MAFIA: Efficient and scalable subspace clustering for very large data sets. *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 443–452, 1999.

[184] Anthony KH Tung, Xin Xu, and Beng Chin Ooi. Curler: finding and visualizing nonlinear correlation clusters. *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 467–478, 2005.

[185] Jiong Yang, Wei Wang, Haixun Wang, and Philip Yu. Δ-Clusters: Capturing Subspace Correlation in a Large Data Set. *Proceedings of the 18th International Conference on Data Engineering*, pages 517–528, 2002.

[186] Chun-Hung Cheng, Ada Waichee Fu, and Yi Zhang. Entropy-based subspace clustering for mining numerical data. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 84–93, 1999.

[187] Gene-Wei Li and X Sunney Xie. Central dogma at the single-molecule level in living cells. *Nature*, 475(7356):308–315, July 2011.

[188] Ben D MacArthur, Avi Ma'ayan, and Ihor R Lemischka. Systems biology of stem cell fate and cellular reprogramming. *Nature reviews. Molecular cell biology*, 10(10): 672–681, October 2009.

[189] W Shi, H Wang, G Pan, Y Geng, Y Guo, and D Pei. Regulation of the Pluripotency Marker Rex-1 by Nanog and Sox2. *Journal of Biological Chemistry*, 281(33):23319–23325, August 2006.

[190] Violetta Karwacki-Neisius, Jonathan Göke, Rodrigo Osorno, Florian Halbritter, Jia Hui Ng, Andrea Y Weiße, Frederick C K Wong, Alessia Gagliardi, Nicholas P Mullin, Nicola Festuccia, Douglas Colby, Simon R Tomlinson, Huck-Hui Ng, and I Chambers. Reduced Oct4 expression directs a robust pluripotent state with distinct signaling activity and increased enhancer occupancy by Oct4 and Nanog. *Cell Stem Cell*, 12(5):531–545, May 2013.

[191] Joshua S Marcus, W French Anderson, and Stephen R Quake. Microfluidic Single-Cell mRNA Isolation and Analysis. *Analytical chemistry*, 78(9):3084–3089, May 2006.

[192] Sameer S Bajikar, Christiane Fuchs, Andreas Roller, Fabian J Theis, and Kevin A Janes. Parameterizing cell-to-cell regulatory heterogeneities via stochastic transcriptional profiles. *Proceedings of the National Academy of Sciences*, 111(5):E626–35, February 2014.

[193] Zach Hensel, Haidong Feng, Bo Han, Christine Hatem, Jin Wang, and Jie Xiao. Stochastic expression dynamics of a transcription factor revealed by single-molecule noise analysis. *Nature structural & molecular biology*, 19(8):797–802, August 2012.

[194] Victoria Moignard, Iain C Macaulay, Gemma Swiers, Florian Buettner, Judith Schütte, Fernando J Calero-Nieto, Sarah Kinston, Anagha Joshi, Rebecca Hannah, Fabian J Theis, Sten Eirik Jacobsen, Marella F de Bruijn, and Berthold Gottgens. Characterization of transcriptional networks in blood stem and progenitor cells using high-throughput single-cell gene expression analysis. *Nature cell biology*, 15(4): 363–372, April 2013.

[195] H R Hulett, W A Bonner, J Barrett, and L A Herzenberg. Cell Sorting: Automated Separation of Mammalian Cells as a Function of Intracellular Fluorescence. *Science*, 166(3906):747–749, November 1969.

[196] A Tomer, L A Harker, and S A Burstein. Purification of human megakaryocytes by fluorescence-activated cell sorting. *Blood*, 70(6):1735–1742, December 1987.

[197] P Malatesta, E Hartfuss, and M Götz. Isolation of radial glial cells by fluorescent-activated cell sorting reveals a neuronal lineage. *Development*, 127(24):5253–5263, December 2000.

[198] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, June 2010.

[199] Y Ann Chen, Jonas S Almeida, Adam J Richards, Peter Müller, Raymond J Carroll, and Baerbel Rohrer. A nonparametric approach to detect nonlinear correlation in gene expression. *Journal of computational and graphical statistics : a joint publication of American Statistical Association, Institute of Mathematical Statistics, Interface Foundation of North America*, 19(3):552–568, September 2010.

[200] Dag Tjøstheim and Karl Ove Hufthammer. Local Gaussian correlation: A new measure of dependence. *Journal of Econometrics*, 172(1):33–48, 2013.

[201] R L F Cordeiro, A J M Traina, C Faloutsos, and C Traina. Halite: Fast and Scalable Multiresolution Local-Correlation Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 25(2):387–401.

[202] Juliane Schäfer and Korbinian Strimmer. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical applications in genetics and molecular biology*, 4(1), 2005.

[203] J Hasenauer, Verena Wolf, A Kazeroonian, and F J Theis. Method of conditional moments (MCM) for the Chemical Master Equation. *Journal of Mathematical Biology*, 69(3):687–735, August 2013.

[204] Pavol Bokes, John R King, Andrew T A Wood, and Matthew Loose. Multiscale stochastic modelling of gene expression. *Journal of Mathematical Biology*, 65(3): 493–520, October 2011.

[205] Christopher K R T Jones. Geometric singular perturbation theory. In *Dynamical Systems*, pages 44–118. Springer Berlin Heidelberg, 1995.

[206] P A Lagerstrom. *Matched Asymptotic Expansions*, volume 76 of *Ideas and Techniques*. Springer Science & Business Media, New York, NY, March 2013.

[207] Peter S Swain, Michael B Elowitz, and Eric D Siggia. Intrinsic and extrinsic contributions to stochasticity in gene expression. *Proceedings of the National Academy of Sciences*, 99(20):12795–12800, October 2002.

[208] Jonathan W Young, James C W Locke, Alphan Altinok, Nitzan Rosenfeld, Tigran Bacarian, Peter S Swain, Eric Mjolsness, and Michael B Elowitz. Measuring single-cell gene expression dynamics in bacteria using fluorescence time-lapse microscopy. *Nature Protocols*, 7(1):80–88, December 2011.

[209] Yuichi Taniguchi, Paul J Choi, Gene-Wei Li, Huiyi Chen, Mohan Babu, Jeremy Hearn, Andrew Emili, and X Sunney Xie. Quantifying E. coli proteome and transcriptome with single-molecule sensitivity in single cells. *Science*, 329(5991):533–538, July 2010.

[210] Milton Abramowitz and Irene A Stegun. *Handbook of Mathematical Functions*. with Formulas, Graphs, and Mathematical Tables. Courier Corporation, April 2012.

[211] Kevin R Sanft, Sheng Wu, Min Roh, Jin Fu, Rone Kwei Lim, and Linda R Petzold. StochKit2: software for discrete stochastic simulation of biochemical systems with events. *Bioinformatics*, 27(17):2457–2458, September 2011.

[212] M Galassi. GNU Scientific Library Reference Manual, 3 edition.

[213] N Michel and M V Stoitsov. Fast computation of the Gauss hypergeometric function with all its parameters complex with application to the Pöschl–Teller–Ginocchio potential wave functions. *Computer Physics Communications*, 178(7):535–551, April 2008.

[214] Christopher Kormanyos. Algorithm 910. *ACM Transactions on Mathematical Software*, 37(4):1–27, February 2011.

[215] Jan Snyman. *Practical Mathematical Optimization.* An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms. Springer Science & Business Media, March 2005.

[216] A P Dempster, N M Laird, and D B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 1977.

[217] David M Suter, Nacho Molina, David Gatfield, Kim Schneider, Ueli Schibler, and Felix Naef. Mammalian genes are transcribed with widely different bursting kinetics. *Science*, 332(6028):472–474, April 2011.

[218] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing*, 22 (5):1009–1020, August 2011.

[219] Jean-Michel Marin, Pierre Pudlo, Christian P Robert, and Robin J Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012.

[220] Carolin Loos, Carsten Marr, Fabian J Theis, and Jan Hasenauer. Approximate Bayesian Computation for Stochastic Single-Cell Time-Lapse Data Using Multivariate Test Statistics. In O Roux and J Bourdon, editors, *Computational Methods in Systems Biology*, pages 52–63, June 2015.

[221] Christian P Robert, Jean-Marie Cornuet, Jean-Michel Marin, and Natesh S Pillai. Lack of confidence in approximate Bayesian computation model choice. *Proceedings of the National Academy of Sciences*, 108(37):15112–15117, September 2011.

[222] Arnaud Doucet and A M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 2009.

[223] R E Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME-Journal of Basic Engineering Series D*, pages 35–45, 1960.

[224] Brian D O Anderson and John B Moore. *Optimal Filtering*. Courier Corporation, May 2012.

[225] Christophe Andrieu, Arnaud Doucet, and Roman Holenstein. Particle Markov Chain Monte Carlo for Efficient Numerical Simulation. In *Monte Carlo and Quasi-Monte Carlo Methods 2008*, pages 45–60. Springer Berlin Heidelberg, 2009.

[226] N J Gordon, D J Salmond, and AFM Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F (Radar and Signal ...*, 140 (2):107, 1993.

[227] Darren J Wilkinson. *Stochastic Modelling for Systems Biology, Second Edition*. CRC Press, November 2011.

[228] Adrian F M Smith, Alan E Gelfand, and American Statistical Association. Bayesian Statistics Without Tears, 1992.

[229] Ben Calderhead and Mark A Girolami. Estimating Bayes factors via thermodynamic integration and population MCMC. *Computational Statistics & Data Analysis*, 53 (12):4028–4045, October 2009.

[230] Aleksandra M. M Walczak, Andrew Mugler, and ChrisH Wiggins. Analytic Methods for Modeling Stochastic Regulatory Networks. In Xuedong Liu and Meredith D Betterton, editors, *Methods in Molecular Biology*, pages 273–322. Humana Press, 2012.

[231] Justin Feigelman, Nikola Popović, and Carsten Marr. A case study on the use of scale separation-based analytical propagators for parameter inference in models of stochastic gene regulation . *Journal of Coupled Systems and Multiscale Dynamics*, pages 1–11, April 2015.

[232] Olivia Padovan-Merhar, Gautham P Nair, Andrew G Biaesch, Andreas Mayer, Steven Scarfone, Shawn W Foley, Angela R Wu, L Stirling Churchman, Abhyudai Singh, and Arjun Raj. Single Mammalian Cells Compensate for Differences in Cellular Volume and DNA Copy Number through Independent Global Transcriptional Mechanisms. *Molecular cell*, 58(2):339–352, April 2015.

[233] Michael K Pitt and Neil Shephard. Filtering via Simulation: Auxiliary Particle Filters. *Journal of the American Statistical Association*, 94(446):590–599, June 1999.

[234] Carsten Marr, Michael Schwarzfischer, Justin Feigelman, Simon Hastreiter, Philipp Hoppe, Dirk Loeffler, Konstantinos Kokkaliaris, Max Endele, Bernhard Schauberger, Oliver Hilsenbeck, J Hasenauer, Konstantinos Anastassiadis, Fabian J Theis, Timm Schroeder, and Adam Filipczyk. Network Plasticity of Pluripotency Transcription Factors in Embryonic Stem Cells. November 2013.

[235] Elsa Abranches, Evguenia Bekman, and Domingos Henrique. Generation and Characterization of a Novel Mouse Embryonic Stem Cell Line with a Dynamic Reporter of Nanog Expression. *PLoS One*, 8(3):e59928, March 2013.

[236] Michael Schwarzfischer. *Quantification and analysis of single-cell protein dynamics in stem cells using time-lapse microscopy.* PhD thesis, Technical University of Munich, 2014.

[237] Michael Strasser, Fabian J Theis, and Carsten Marr. Stability and Multiattractor Dynamics of a Toggle Switch Basedon a Two-Stage Model of Stochastic Gene Expression. *Biophysical journal*, 102(1):19–29, January 2012.

[238] Jonathan R Chubb, Tatjana Trcek, Shailesh M Shenoy, and Robert H Singer. Transcriptional pulsing of a developmental gene. *Current Biology*, 16(10):1018–1025, May 2006.