In silico modelling for refining gene regulatory network inference

Thomas Hillerton
In silico modelling for refining gene regulatory network inference

Thomas Hillerton

Academic dissertation for the Degree of Doctor of Philosophy in Biochemistry towards Bioinformatics at Stockholm University to be publicly defended on Friday 27 October 2023 at 14.00 in Air and Fire, SciLifeLab, Tomtebodavägen 23A.

Abstract
Gene regulation is at the centre of all cellular functions, regulating the cell's healthy and pathological responses. The interconnected system of regulatory interactions is known as the gene regulatory network (GRN), where genes influence each other to maintain strict and robust control. Today a large number of methods exist for inferring GRNs, which necessitates benchmarking to determine which method is most suitable for a specific goal. Paper I presents such a benchmark focusing on the effect of using known perturbations to infer GRNs.

A further challenge when studying GRNs is that experimental data contains high levels of noise and that artefacts may be introduced by the experiment itself. The LSCON method was developed in paper II to reduce the effect of one such artefact that can occur if the expression of a gene shows no or minimal change across most or all experiments.

With few fully determined biological GRNs available, it is problematic to use these to evaluate an inference method's correctness. Instead, the GRN field relies on simulated data, using a known GRN and generating the corresponding data. When simulating GRNs, capturing the topological properties of the biological GRN is vital. The FFLatt algorithm was developed in paper III to create scale-free, feed-forward loop motif-enriched GRNs, capturing two of the most prominent topological features in biological GRNs.

Once a high-quality GRN is obtained, the next step is to simulate gene expression data corresponding to the GRN. In paper IV, building on the FFLatt method, an open-source Python simulation tool called GeneSNAKE was developed to generate expression data for benchmarking purposes. GeneSNAKE allows the user to control a wide range of network and data properties and improves on previous tools by featuring a variety of perturbation schemes along with the ability to control noise and modify the perturbation strength.

Keywords: Gene regulatory networks, simulation, benchmarking, method development.

Stockholm 2023
http://urn.kb.se/resolve?urn=urn:nbn:se:su:diva-221155

ISBN 978-91-8014-504-6

Department of Biochemistry and Biophysics

Stockholm University, 106 91 Stockholm
IN SILICO MODELLING FOR REFINING GENE REGULATORY NETWORK INFERENCE

Thomas Hillerton
In silico modelling for refining gene regulatory network inference

Thomas Hillerton
Abstract

Gene regulation is at the centre of all cellular functions, regulating the cell's healthy and pathological responses. The interconnected system of regulatory interactions is known as the gene regulatory network (GRN), where genes influence each other to maintain strict and robust control. Today a large number of methods exist for inferring GRNs, which necessitates benchmarking to determine which method is most suitable for a specific goal. Paper I present such a benchmark focusing on the effect of using known perturbations to infer GRNs.

A further challenge when studying GRNs is that experimental data contains high levels of noise and that artefacts may be introduced by the experiment itself. The LSCON method was developed in paper II to reduce the effect of one such artefact that can occur if the expression of a gene shows no or minimal change across most or all experiments.

With few fully determined biological GRNs available, it is problematic to use these to evaluate an inference method's correctness. Instead, the GRN field relies on simulated data, using a known GRN and generating the corresponding data. When simulating GRNs, capturing the topological properties of the biological GRN is vital. The FFLatt algorithm was developed in paper III to create scale-free, feed-forward loop motif-enriched GRNs, capturing two of the most prominent topological features in biological GRNs.

Once a high-quality GRN is obtained, the next step is to simulate gene expression data corresponding to the GRN. In paper IV, building on the FFLatt method, an open-source Python simulation tool called GeneSNAKE was developed to generate expression data for benchmarking purposes. GeneSNAKE allows the user to control a wide range of network and data properties and improves on previous tools by featuring a variety of perturbation schemes along with the ability to control noise and modify the perturbation strength.
Papers included in the thesis

Paper I: Knowledge of the perturbation design is essential for accurate gene regulatory network inference.
Deniz Seçilmiş, Thomas Hillerton, Andreas Tjärnberg, Sven Nelander, Torbjörn E. M. Nordling and Erik L. L. Sonnhammer. 
Scientific Reports 12, 16531, October 2022. 
https://doi.org/10.1038/s41598-022-19005-x
Contribution: Data generation using GeneSPIDER toolbox, creation of Ridge regression scripts, creation of scripts for accuracy calculation.

Thomas Hillerton, Deniz Seçilmiş, Sven Nelander, Erik L L Sonnhammer. 
Bioinformatics, Volume 38, Issue 8, March 2022. 
https://doi.org/10.1093/bioinformatics/btac103
Contribution: Creation of LSCON algorithm, testing of algorithm and writing of the paper.

Paper III: Generation of Realistic Gene Regulatory Networks by Enriching for Feed-Forward Loops.
Erik K. Zhivkoplias† Oleg Vavulov† Thomas Hillerton Erik L.L. Sonnhammer 
Frontiers in Genetics, 13, 815692, February 2022. 
https://doi.org/10.3389/fgene.2022.815692
Contribution: Creation of scripts for stability analysis, helped E.K Zhivkoplias with writing of the paper.

Thomas Hillerton, Erik K. Zhivkoplias, Mateusz Garbulowski and Erik L. L. Sonnhammer 
Contribution: Researched and implemented ODE model for gene expression simulation, integrated the FFLatt algorithm and other python tools, tested the various tools, and writing of the paper.
## Contents

1. Introduction .................................................................................................................. 3
2. Gene regulation ............................................................................................................. 5
3. Gene regulatory networks ........................................................................................... 8
   3.1 The gene regulatory network in biology ............................................................... 8
   3.2 Capturing the gene regulatory network in a model ............................................. 9
   3.3 The GRN model, a tool for overseeing and understanding the cell ............... 11
   3.4 Noise - a property of biological data ................................................................. 12
   3.5 Complex systemic behaviours in gene regulatory networks ......................... 16
      3.5.1 The feed-forward loop .............................................................................. 17
      3.5.2 Autoregulation ......................................................................................... 19
4. Simulating gene regulatory networks ......................................................................... 21
   4.1 Modelling gene expression through differential equations ............................ 23
   4.2 Using linear approximation to reduce model complexity .............................. 25
5. Gene regulatory network inference ........................................................................... 27
   5.1 Selecting the right GRNI method ..................................................................... 29
   5.2 Utilising least squares in GRNI ...................................................................... 30
      5.2.1 Least squares with cutoff ......................................................................... 30
      5.2.2 Least squares with cutoff normalised ..................................................... 31
      5.2.3 Penalised least-squares regression methods .......................................... 32
   5.3 Non-perturbation-based methods ................................................................. 34
6. Present investigations ................................................................................................. 35
   6.1 Knowledge of the perturbation design is essential for accurate gene regulatory network inference (Paper I) ................................................................................. 35
   6.2 Fast and accurate gene regulatory network inference by normalised least squares regression (paper II) ................................................................................. 36
   6.3 Generation of Realistic Gene Regulatory Networks by Enriching for Feed-Forward Loops (paper III) ............................................................................... 37
   6.4 GeneSNAKE: a Python package for benchmarking and simulation of gene regulatory networks and expression data (paper IV) ....................................... 38
7. Discussion and concluding remarks ........................................................................ 40

Populärvetenskaplig sammanfattning .......................................................................... 42
Acknowledgment .......................................................................................................... 43
References ..................................................................................................................... 44
1 Introduction

A cell can be viewed as a system of complex molecular interactions, with many different proteins, metabolites, RNA molecules, and myriads of other molecules continuously binding and unbinding to each other. These molecular interactions serve as each cellular function's root and strict regulators. At the centre of this system sits gene regulation. By selectively expressing the correct set of genes at the right time, complex functions can be modulated or activated to respond to the environment surrounding the cell. The selective nature of gene expression holds true both in simple single-cell organisms as well as complex multicellular organisms. Gene regulation gives cells a wide array of responses to internal and environmental stimuli. Today these regulatory interactions are well studied both on a chemical and functional level. However, a comprehensive map of how various genes affect each other has yet to be found. Creating such a map has long been a central focus in systems biology [1–3].

This work centres around the field of gene regulatory network inference (GRNI). The field studies the complex web of interactions that makes up gene regulation within the genome of a cell, with the end goal of creating a representational map of the system while ensuring it remains comprehensible. GRNI primarily focus on determining the effect of a given gene on the expression of other genes while leaving the exact nature of the interaction undetermined [4]. In this way, a gene regulatory network (GRN) can be considered a map of what interactions exist but cannot be used to determine exactly how the genes interact. As GRNI focuses on changes in expression, the usual way to capture GRNs is to perturb one or more genes in the system and then measure the change in gene expression. The perturbation of the system is of particular importance as only by introducing some form of change can the relationship between the genes be detected [5,6]. However, capturing the relationship between genes is a complicated task, partly due to the noisy nature of the currently available methods for detecting gene expression and partly due to the complicated nature of these relationships. As genes do not interact physically with one another, in the way proteins or metabolites often bind to each other when interacting, or even in a direct manner, the interaction potentially being mediated through several other molecular interactions. For example, a gene may code for a protein that act as a small part of a protein complex that will, in turn, degrade a set of mRNA molecules, creating an inhibitory effect without the first gene having any explicit binding or targeting of the inhibited gene. In addition, due to the importance of gene expression, many regulations are part of intricate webs with multiple genes regulating each other or jointly regulating a set of other genes [7], making each separate regulatory effect hard to unravel.
This difficulty has led to the development of many methods aimed at reconstructing an accurate GRN from gene expression data. These methods rely on various algorithms and mathematical models, each with its own strengths and weaknesses [8]. So far, no clear best method or method group exists, and the choice of method should vary based on the available data and the goal of the GRN inference. The strength and weaknesses of some of these methods are reviewed in paper I. Regardless of the approach used to find the GRN, one of the most significant issues is the noise in the data masking much of the perturbation's effect [9]. The issue arises from unexplainable variance, often caused by stochasticity in the molecular methods used for measuring gene expression, masking the effect in the system after the perturbation. Noise causes slight changes in expression to look significant or obfuscates actual change making it impossible to reconstruct the GRN [10]. Along with noise other experimental artefacts such as the expression of some genes not being accurately quantified or expression in essential genes not changing due to cellular mechanisms compensating for the reduced expression. A work for addressing some of the potential effects of these artefacts on GRNI is presented in paper II.

Another key concern in developing and benchmarking novel methods for GRNI is what data is used for evaluating performance. To accurately determine how well a method can reconstruct a GRN from expression data, a known GRN is needed [10,11]. A known GRN is one where all interactions are known, and corresponding effects can be detected in gene expression data. However, data corresponding to a known GRN is almost impossible to obtain from experimental data as only some parts of the total GRN are entirely known today. Additionally, each cell will have differences in its GRN based on the cell type [12], cell stage [13] and health status [14]. As such, the GRNI field has taken to rely on simulated data; data created computationally to match experimental data well enough to replace it in benchmarking. Many different tools for this have been developed. However, the tools tend to struggle with capturing the complex reality of the genetic interactions or missing crucial features that allow the user to simulate the data required for their study. Expanding the possibility of simulations and approximating biological GRNs are discussed in papers III and IV. Paper III showed a clear improvement in the creation of GRN models, focusing on recurring functional groups, motifs, within the GRN. Paper IV builds on that work to deliver a fully functional simulation and evaluation tool called GeneSNAKE. That allows users to reliably create data and evaluate GRNI predictions.
2 Gene regulation

Gene regulation has been extensively studied in a systems biology context, a mechanistic and chemical context, and on a more detailed scale looking at direct gene-gene interactions. Gene regulation is linked to all cellular functions through protein and non-coding RNA expression, changes in gene expression being the primary way cells can respond to internal or external changes [15]. With gene regulation so closely tied to all functions in a cell, it follows that the gene regulatory system itself is tightly moderated and robust with multiple levels of control. The gene regulatory system can be roughly divided into three levels, (I) short-term transcriptional control, which is well described in both prokaryotic and eukaryotic cells and consists of proteins, transcription factors, which bind and unbind to the DNA. The regulation is comparatively direct with transcription factors binding to cis-regulatory elements present in the DNA structure. Once bound the transcription factors regulate gene expression by either activating the gene by recruiting RNA-polymerases and associated factors, expressing the gene, or inhibiting the gene by preventing the binding of activating factors or RNA-polymerases [16].

(II) Epigenetic regulation is similar to transcriptional control in that it primarily regulates genes on a transcriptional level. It does this in a way that focuses on long-term regulation, often changes in the expression can last between hours to years. Epigenetic regulation classically is considered to consist of DNA-methylation, chromatin remodelling and noncoding RNA based gene silencing. A good example of this is the genomic imprinting found in diploid cells, where alleles will be semi-permanently silenced through DNA methylation during embryogenesis [17]. Epigenetic regulation has primarily been studied in eukaryotic cells, however, some evidence exists for similar regulation existing in prokaryotic cells, primarily in the form of DNA-methylation [18].
Figure 1: The various levels of gene regulation. Gene regulation occurs on all levels of the traditional model of the central dogma. The initial regulation consists of transcription-based regulation, either through transcription factors or epigenetic regulation. Following transcription, several processes exist to modify or degrade RNA molecules. For example, in eukaryotic cells, RNA molecules are spliced to contain the correct exons before being guided out of the nucleus. Outside the nucleus, the selection of RNA molecules continues with, for example, miRNA-guided cleaving of selected mRNAs.

(III) Post-transcriptional regulation, in turn, consists primarily of various processes targeting the transcribed RNA molecule. Typical forms of this regulation are RNA-processing such as splicing, RNA-transport and localization control, translational control, and targeted RNA-degradation. A good example of the last two are targeted degradation guided by miRNA. The miRNA molecule will bind to a protein complex known as the RNA-induced silencing complex and bind to mRNA molecules. mRNA molecules which have a significant base-pairing with the miRNA will then be cleaved, removing the poly-A tail causing exonucleases to degrade the mRNA [19]. Post-transcriptional regulation along with transcription factor mediated regulation are fast acting and respond on a time scale of minutes to hours [15], allowing the cell to respond rapidly to environmental changes.
These different forms of regulation intertwine to ensure that the right genes are expressed at the right time and equally important, that the wrong genes are not expressed at the wrong time or place. The system gives a fine-grain and constant control over gene expression, allowing cells to perform specific and complicated functions in response to their environment. In addition to these active forms of regulation, the system contains a passive regulatory effect where most genes can only be expressed under the direct influence of an activating transcription factor [15]. While this might seem obvious when looking at it on a single gene, from a systemic perspective, it plays a vital role in allowing cells to limit inhibitory signals while still letting the cell return to a steady state through the passive degradation of proteins and RNA. This passive regulation reflects a central issue for the cell when it comes to gene regulation, that each signal in the system comes at a cost for the cell as genes must be transcribed, proteins translated and molecules transported and activated, each step either tying up molecules preventing them from being used elsewhere, and expending energy. This passive regulation reflects how important gene regulation is, and how to understand gene regulation it must be studied on a systemic scale. Many of the gene regulatory functions only truly make sense when viewed from a systemic perspective.
3 Gene regulatory networks

Looking at gene regulation from a systemic view quickly becomes highly complex, especially when all molecular interactions are considered in some detail. The gene regulatory network (GRN) presents a solution to this. GRNs is a tool that relies on a network-based representation of the system that can be used to map the interactions while abstracting complex molecular interactions. However, it is important to remember that, unlike many other biological networks, a GRN is an existing system with each interaction existing on the cell. The term GRN is used interchangeably between the biological GRN existing in the cell and the model GRN. The two types of GRNs are highly intertwined, with the model GRN giving a network-based overview of the biological GRN without fully accounting for the complex dynamics of the molecular system. To fully account for the molecular system is, however, unnecessary and may, in many cases, be counterproductive for the model to be informative. In this work, the model of the system will be in focus, but to understand what it represents, the cellular GRN will be given a brief introduction.

3.1 The gene regulatory network in biology

At a cellular level, a GRN is a time-delayed probabilistic system with molecules binding and unbinding to each other making up the regulation. This binding is primarily controlled by kinetic binding probability, the concentration of and affinity to the target determining the regulatory molecule's effect on the target. The system is structured and tends to be hierarchical, where a small number of regulators will activate larger sets of functionally related genes. In humans and many other multicellular life forms, the system can be divided into (I) master regulators, genes that act as activating transcription factors for large sets of genes. (II) secondary regulators, genes which are activated by the master regulators and, in turn, tend to activate a (III) group of genes that take part in the same or highly related functions. While the hierarchical division is not universally applicable, and all gene regulatory pathways do not follow the same strict hierarchy. The hierarchy is common enough to be seen in the topology in the GRN, which takes on a scale-free nature [20,21]. A scale-free network is one where most nodes have very few edges, with a limited number of nodes containing a large number. The probability of finding a node with a given number of edges diminishes as the number of edges increases.
As a general rule, the probability of a node having \( k \) connections \( P(k) \) should follow a power-law distribution given by:

\[
P(k) \sim k^{-\gamma}
\]

Where \( \gamma \) is a parameter controlling how quickly the probability of finding a node with \( k \) edges will decrease as \( k \) grows [22]. \( \gamma \) is usually between 2 and 4 but can, in theory, grow towards infinity. GRNs follow this distribution asymptotically but may not show a perfect fit to the expected curve described by the equation [20].

For cells to survive and function, the GRN must be highly robust. The GRN, therefore, contains several features explicitly aimed at ensuring correct regulation. The most direct of these features is that the entire regulatory space is encoded directly in the genome, with molecular interaction only being possible at predetermined matching places [23]. Compared to many other regulatory systems, such as traffic flow or electrical circuits, this limits the possibility of faulty signals. In addition, the GRN contains several structures aimed at ensuring that erroneous signals result in only minor changes in the GRN. These structures consist of topological features in the GRN, forming functional groups containing genes that together act as brakes on expression or limit activation to require multiple input signals or consistent signals over time [24]. The functional groups and their roles are described in greater detail in section 3.5 Complex systemic behaviours in gene regulatory networks. Part of the reason the GRN must be this robust is that it must act in a highly deterministic way, something that can be observed in biology, where the effect of a given stimulus generally has a consistent response across different cells or time points. The deterministic nature is evident in the developmental stages of an organism where various tissues will form in a discrete and highly distinguished manner, e.g., an arm will never be a head and vice versa. A feat primarily guided by the gene regulation in each cell involved in this process [25].

### 3.2 Capturing the gene regulatory network in a model

As seen in this brief introduction, the biological GRN is central to understanding the cell and is a decidedly challenging problem to grasp in its entirety. It would be almost impossible to create a model that could capture every detail of the system and still offer an understandable overview of the system. Instead, a model that captures just enough of the details to allow them to be represented is needed. The GRN model fulfils this requirement by focusing on the function of the biological GRN, capturing regulatory relationships only as functional...
The GRN model focuses only on increasing or decreasing RNA expression rather than the details of the molecular relationships. The reduction in details allows the GRN model to act more as a map of the regulation. This map allows researchers to quickly find which genes regulate each other and their regulatory relationships without needing a detailed understanding of the exact molecular processes involved. The GRN model consists of a network or graph, with the terms benign used interchangeably, that captures the regulatory relations in a direct representation of the system. The network consists of nodes representing the genes and edges representing the regulatory interactions. These edges often contain detailed information about the interaction, primarily in the form of the sign and occasionally the strength of the regulatory interaction. The sign of the edge corresponds to the nature of the interaction with the regulator either activating, a positive sign, or inhibiting, a negative sign, the target. In graphical representations, the sign is usually presented as an arrow for activation and a bar indicating inhibition. In addition to the sign, the strength of the interaction can be represented by weight, a numerical value between plus/minus infinity, with a higher absolute weight indicating a stronger interaction regardless of the sign. An example of this type of model is given in figure 2.
A fundamental difference between the GRN model and most other biological network models is that the GRN is directed, meaning there is a known and specific direction in the information flow through the network. A regulator always passes information to the target, while information often does not flow directly back to the regulator [23]. For this reason, the edges are encoded as arrows even if the sign is unknown and expanded to include arrows and bars if the sign is known. Overall, this directionality is the key to a GRN, as only with it can regulatory relationships be determined.

![Figure 3: The components of a GRN.](image)

A GRN model consists of nodes, as circles representing the genes, and edges, as lines representing the regulatory interactions. The edges further indicate the type of regulation, with activation shown as arrows and inhibition shown as bars.

### 3.3 The GRN model, a tool for overseeing and understanding the cell

The GRN model, with its simplified information, offers beneficial insight into the cell's core. The information in the model allows researchers to understand a cell's functions and potential paths to alter these functions. A key example of the possibility of the GRN model were given by Gerstein et al 2012 who suggested that due to the hierarchical nature of the GRN targeting interventions against transcription factors that act as go-betweens between the master regulators and the functional groups can offer a more substantial effect while limiting side-effects [23]. Selecting regulators that are less central in the GRN is explained by the master regulators are often involved in multiple functions.
and vital to the cell's survival. The master regulators, thus, make for poor targets for any intervention due to the risk of disrupting additional functions. Similarly, while the genes in the functional group are vital for the execution of the function, they often only correspond to a fraction of the effect. An intervention on that level may be far less effective than one on a higher level where the functional group is activated. By knowing the structure and flow of information through the GRN, researchers can, through this kind of relationship, determine with great accuracy where to intervene, be it for the purpose of developing treatments or to study the cell itself further [30–32].

The same information is also a vital step in personalised medicine, allowing for specific targeting of drugs [33] and the reliance on functional pathways as biomarkers in complex diseases such as cancer. Cancer hallmarks have been shown to involve pathways with several possible fault points [34], suggesting that activity in the broader GRN may act as a better marker for both cancer and potential treatment than the activity in any single gene can do alone. In a broader view, the centrality of the GRN leads the model GRN to play a central role in understanding cells in general. As previously mentioned, an area where GRN research has played a prominent role is developmental biology. Gene regulation, and especially correct gene regulation, plays a crucial and active role during the development of both animals and plants. The GRN model offers vital insight into the complex pattern of gene regulation that occurs throughout the developmental process. Strict gene regulation ensures that each tissue and cell type develops in the right place and time [35]. The GRN further offers insight into specific genes and their regulation and explains many functional effects observed in gene regulation. Many of the effects crucial to foetal development originate in GRN motifs. For example, the positive autoregulatory motif is strongly linked to foetal development, where it acts as a switch by maintaining gene expression until a strong inhibitory signal is given [36,37]. The positive autoregulatory motif is described in more detail in section 3.5.2 Autoregulation.

3.4 Noise - a property of biological data

Noise is an intrinsic property of biological systems and the experimental methods used to measure these systems [38]. When talking about noise in the context of GRNs, it is really variance that cannot be explained by the experiment that is the issue. Regardless of if this variance is from the natural fluctuation of gene expression, something that has been observed between cells, or due to errors in the measurement, the effect is similar as it limits how well a GRNI method can find regulatory relationships. As the noise affects all genes, it can often mask changes brought on by the regulation or make non-regulated genes appear to change significantly. As it is usually impossible to determine if the
noise is of biological or methodological origin, the two are usually approached as a single issue where both are assumed to obfuscate the experimental results. Despite this, the biological noise could, in theory, contain important information about gene expression. Experiments aimed at capturing and separating the biological noise from methodological noise are usually complex and require advanced experimental setups where ascertaining the noise is the main goal. While this has been done and has provided vital information about how variance in gene expression looks across cells [39–41], it is often not feasible to combine with studies of the GRN due to the increased number of tests that would be needed.

The origin of the noise is often just as unknown as the effect of the noise. However, there are a few common sources. Biological noise originates from several different sources. The most straightforward one is that different cells express genes at different levels due to being at different stages in their cell cycle or other cellular processes. A large part of biological noise can further be attributed to the stochastic behaviour found in all biochemical processes. Some molecules find each other faster or slower simply by random chance [38]. Technical noise, in turn, is often more specific and depends on the method with which the gene expression has been measured. Broadly speaking, it tends to be dependent on the same biochemical stochasticity as in the biological noise, with molecules either binding better to some genes, some molecules binding to the wrong areas of molecules, or some genes being expressed in a lower concentration than others making changes hard to detect accurately [39–41]. When it comes to GRNI, noise is one of the primary concerns when determining whether a method will work. If the effect of the regulatory relationship is masked by noise, the effect caused by perturbing the GRN may become impossible to detect [10,29]. As such, it becomes increasingly challenging to separate true regulatory interactions from the effect of noise as the noise increases. In theory, if the systemic changes in a GRN could be measured without any noise, the reconstruction of most of the system would be trivial [42]. Most regulatory interactions could then be captured through simple methods like linear regression or correlation. With noise, the problem of GRNI becomes much more demanding [29,43], having to rely on methods that can account for the false signal of the noise. Due to the effect of noise on GRNI (see paper I for a detailed study on the effect of noise on GRNI), finding ways to estimate how much noise is in the data is vital before trying to infer GRNs on the data.
With the high impact of noise on GRNI, a central metric in the field is how much noise is in the data. Many methods have been developed for this. However, as the noise is largely unknown, no method can today do more than give a highly approximate estimate. In the works presented here, a method called SNR-L [10] has been extensively used to estimate the noise. SNR-L works by estimating how much signal, the true regulatory relationships, there are compared to noise based on the system's singular values and degrees of freedom.

**Figure 4: Noise across the expression of two genes.** Noise is an unavoidable property of gene expression data, it introduces unexplained variance when measuring expression of the same gene under identical conditions. Some of the noise is biological and can be explained by the stochastic nature of the molecular interactions that regulate gene expression. A second, larger part of the noise comes from the method used to measure expression. In principle, the same molecular stochasticity causes the technical noise as that of the biological. However, the exact cause of either noise type is generally unknown.
SNR-L is defined as:

\[
SNR = \frac{S(Y)}{\sqrt{\chi^2(\alpha, NM)\lambda_Y}}
\]  

(2)

Where \( S \) is the minimum singular value of the \( Y \) matrix containing the relative change. \( \chi^2(\alpha, NM) \) is the chi-square constant for the type 1 error \( \alpha \) with degrees of freedom \( NM \). With \( N \) and \( M \) being the rows, genes, and columns, experiments, of \( Y \), respectively. \( \lambda_Y \) is the variance in \( Y \) [42].

The SNR-L method, however, can often not give exact estimates and especially struggles as the system grows due to the reliance on degrees of freedom. As an alternative to and remedy for the size issue of SNR-L, more straightforward methods for approximating noise have been shown to give reasonably good estimates of how well the data will work for GRNI. Two different methods of this type were developed alongside paper IV.

First, a method referred to as perturbation rank, which uses the relative change in each experiment to estimate how similar the experiment is to the expected perturbation. The method builds on the fact that the gene targeted for perturbation should show the most change. The introduced perturbation is much more potent than any naturally occurring regulation, so the expected change observed should also be far more substantial. Ensuring that an evident perturbation can be observed helps determine that a given experiment has worked but says little about the variation across the data. Second, to capture this system-wide noise, another method is needed, replicate correlation. Replicate correlation measures Pearson correlation between replicates, and by looking at the average correlation between replicates can give a good approximation of how noise the data is. Replicate correlation works under the assumption that each replicate was treated identically, giving a correlation close to one in the absence of noise. Using correlation rather than directly measuring variance between the replicates can be motivated by the fact that all cells are not identical. The difference between cells can result in high variance within the gene expression without any noise. Correlation patterns should, however, remain consistent regardless of the original expression values. To make replicate correlation more robust, it can be compared to the average correlation across all non-replicate experiments. If the correlation is significantly higher than the average, that is strong evidence that the noise does not mask the signal. Note that none of these methods are perfect, and a good value in SNR-L, perturbation effect, and replicate correlation is just an indication that the data may perform well.
3.5 Complex systemic behaviours in gene regulatory networks

In addition to the noise, a major constraint for GRNI is the complexity of the biological GRN. While GRNs are often discussed in terms of target and regulator, two genes at a time, the system often follows far more complex patterns with smaller recurring groups of nodes building up a complex topology. These groups perform important functions that play vital roles on a systemic level [44]. The groups are referred to as motifs, a motif being a reoccurring edge pattern between a set number of nodes in a network [45]. In GRN research, only three-node motifs tend to be considered [46]. The limit of three nodes is explained by the rapid growth of possible combinations as the number of nodes grows. For three-node motifs, 13 possible motifs exist where all nodes are connected. In comparison, for four-node motifs, there exist 199 possible combinations, and for five-node motifs 9,364 combinations exist. The large number of possible combinations makes it almost impossible to determine whether any one motif is significantly enriched in a network. Similarly, two-node motifs are undefined by their very nature, as all nodes would fall into a motif consisting of either one regulator and target or two nodes regulating each other. Somewhat unexpectedly, single-node motifs, where a gene regulates itself, do exist and play a crucial part in the robustness of GRNs [37]. Both three-node and single-node motifs will be described in some detail here as they provide excellent examples of why studying a GRN on a larger scale can give deeper insights into how gene regulation functions. Other important topological patterns exist in the GRN, however, they tend to be far less uniform and can often only be discussed as they relate to a specific regulation. The single input multiple output (SIM) motif is an example of such a pattern. The SIM is a recurring pattern in the GRN where a single regulator controls a set of genes often required for a specific function [47]. However, as the SIM motif can contain any number of nodes, it is far less trivial to define and compare these motifs between networks [48].
3.5.1 The feed-forward loop

While 13 different three-node motifs could exist in a GRN, only one is significantly enriched in GRNs, the Feed-forward loop (FFL). Determining if a motif is important, as opposed to occurring at random, for a GRN is done by comparing the occurrence of all possible FFL motifs across several GRNs to the occurrence of FFLs in random graphs. For these studies, it is assumed that motifs occurring significantly more often than would be expected at random play some role in the GRN. Studies examining the occurrence of three-node motifs in GRNs have repeatedly shown that only the FFL motif is enriched [46,49,50]. The same result was found in paper III, where the FFL motif is the only enriched three-node motif across four species. The FFL motif comprises two regulators and one non-regulator in a pattern where gene 1 regulates gene 2 with both gene 1 and 2 regulating gene 3 [51], an example of an FFL is given in figure 5 and 6. Based on the sign of the regulation, there are eight possible FFL motifs, but only two have been found to be significantly enriched in GRNs, making up about 80% of all FFLs in the *E. coli* GRN [46]. The two

| Figure 5: The 13 possible three-node motifs in GRNs. |
|-------------|-------------|-------------|
| ![Diagram of 13 possible three-node motifs in GRNs](image) |
| In a network, 13 different fully connected three-node motifs exist, which should, if occurring at random, be present with an equal frequency. In a GRN, however, the feed-forward loop, motif 7, is highly enriched in all known GRNs, with a depletion for the other motifs. |
motifs consist of a coherent FFL, where all edges are activating, and an incoherent FFL, where the regulation of the second regulator is inhibitory, the structure of both are shown in figure 6. The clear overrepresentation of these two motifs is indicative of the functional roles these motifs play. Both the coherent and incoherent FFL motifs build on the same principle of molecular expression and degradation. The time it takes for the RNA and protein molecules to be created and destroyed is used in the FFLs to create time-dependent effects within the gene regulation of the target gene.

The coherent FFL has been shown to fill two similar roles regarding the robustness of the GRN. The difference in the function depends on whether both regulators are needed to activate the target gene or if either one can activate the target alone. When both regulators are needed to activate the target, the coherent FFL motif acts as a time-delayed on-switch where the target gene will only be activated if the initial signal on the first regulator persists long enough for the second regulator to reach a critical concentration. If, on the other hand, either regulator can activate the target alone, the FFL motif acts as a time-delayed off-switch. The target in this type of FFL will remain active unless the inhibition of the first regulator persists long enough for the concentration of both regulators to go below the activation threshold. The coherent

![Coherent and Incoherent FFL](image)

**Figure 6: Coherent and incoherent FFL.** In GRNs, two FFL motifs are particularly enriched. One is a coherent FFL where all edges are activating. The other is an incoherent FFL where the second regulator, Gene 2, acts as an inhibitor on the regulated target, Gene 3.
FFL motif thus assures that critical genes are not accidentally turned on or off. Usually, this type of motif will be associated with genes where the cost of activation or deactivation is high for the cell [46,52].

The incoherent FLL, in turn, tend to be linked to genes that the cell needs for rapid responses to changes in the environment or genes where the expression should be a quick pulse. For genes with a rapid response, the first regulator exerts a strong activation on the target gene, leading to a fast rise in its concentration. Once the second regulator is activated, the inhibitory effect will gradually break the initial rapid activation as the concentration of both the target and second regulator increases. Ultimately, the target gene will reach a steady expression level balanced by the two regulators. The breaking effect of the second regulator allows for this rapid response without over-saturating the cell with the target gene and risking an overcorrection to the initial stimuli. For genes with a pulsating expression, the first regulator is either quickly turned off after activation or the inhibition of the second regulator is stronger than the activation. With this setup of activating the first gene, the target gene will briefly express an RNA and then be silenced by the second regulator [46,52].

3.5.2 Autoregulation

While the FFL is the only known enriched three-node motif in GRNs, there are two single-node motifs, the negative and positive autoregulatory motifs that are enriched in GRNs [53]. The autoregulatory motif consists of a single where the gene will either inhibit or activate its own transcription. The negative autoregulatory motif is the more common of the two and consists of a gene that inhibits its own transcription. The motif has primarily been found in transcription factors and is believed to be linked to two important functions. First, it allows for a similar increase in the initial transcription activation as the incoherent FFL, with self-inhibition playing a similar role as the second regulator in the FFL motif. The gene inhibiting its own transcription allows for a rapid increase in transcription rate upon activation, with the self-inhibition acting as a break once it reaches a high enough concentration [37]. Without this type of break function in place, the response would either have to be much lower or risk the concentration of the transcription factor rapidly growing out of control. Secondly, and perhaps even more vital on a systemic level, is the ability of the negative autoregulatory effect to create a robust expression level of the gene around a steady state. The self-inhibition here plays a role in correcting for small fluctuations in the gene's expression, both breaking and driving transcription of the gene. Small fluctuations in the genes can occur due to changes in the activator available or stochasticity in the binding of activators. However, the self-inhibition will respond to this fluctuation by either reducing or increasing inhibition as the concentration of the gene goes up and
down. In this way, the self-inhibition creates a self-regulating drive to a pre-determined concentration limiting fluctuations of the gene [54]. An additional advantage is that due to the self-regulating drive towards a specific concentration, only a strong signal can move the gene expression into a new state, higher or lower expression. This balance allows the system to express genes important for continuous functions at a consistent level. It also allows for genes to act both as house-keepers and have a function to change the cell state, as only a strong signal will move the transcription away from the balanced state, preventing accidental changes in the cell state [37].

While less common, the positive autoregulatory motif still plays a crucial role. The positive autoregulatory motif consists of a single gene that activates itself, increasing or maintaining its own expression rate. When comparing the two autoregulatory motifs, the positive autoregulatory motif can, at first, seem counterintuitive to the robust nature of the GRN. Self-activation has the opposite effect to self-inhibition, causing a slower and more stochastic expression in the gene. However, self-activation offers one vital feature for the GRN, it gives the gene a form of bistability. When not active, the gene will remain unexpressed, but on activation, the gene can maintain a high level of expression without additional signals, only ending with a strong inhibitory signal [37]. This type of bistability is a crucial feature in cellular processes where the cell should not be able to return to the previous stage. For example, the positive autoregulatory motif plays a crucial role during development, where it is vital that a cell cannot go back to an earlier stage once it starts differentiating towards a specific cell type [31,55].

In model GRNs, one more type of autoregulation can be found in the form of the passive RNA degradation that constantly occurs in a cell. While not strictly a regulatory interaction, the probabilistic RNA and protein degradation in the cell will lead to an increased degradation as a gene's expression increases. This type of autoregulation can have important implications on the GRN model [42] in that it ensures that expression in the model does not grow infinitely. This passive autoregulation is often essential in GRN models, and it is important to know about it when studying GRN models. For example, the GeneSPIDER model [10] used in paper I, and II rely heavily on passive autoregulation to ensure the GRN can reach a steady state. However, whether this is a regulatory interaction, and should be included in the GRN, can be debated, as the degradation rate is not directly regulated by the gene being expressed. Due to this uncertainty in how to classify this type of autoregulation and the universal nature of the effect, this autoregulation is often not included in inferred GRNs. The effect is instead implicitly in the system as all mRNA molecules will degrade over time.
Obtaining data for developing and evaluating methods is a significant challenge in GRNI. Since today no gene expression data that corresponds to a fully determined GRN exists using data obtained from experiments for testing novel methods have been largely unsuccessful with most methods scoring low on correctness [56]. Unknown interactions in the GRN make it impossible to determine if an edge predicted by a GRNI method is false or not yet discovered. To remedy this issue, the GRN field relies on simulated GRNs with corresponding data [10,11]. These simulations either use a known GRN as input or create an artificial GRN, mirroring the properties of the biological GRN, and then create data based on the dynamics in that GRN. Regardless of the origin of the GRN, the data is created to fit this exact known GRN, removing the risk of unknown interactions. Methods can then be evaluated based on how well they can reconstruct the known GRN, making it trivial to determine true and false edges recovered [29].

Today, several tools exist to perform these simulations [10,11,57–60], though most are not widely used. The tools rely on several approaches to capture the dynamics of the GRN, the most popular being based on ordinary differential equations (ODE) [11]. Here two of the underlying approaches will be described in some detail, ODE models and linear approximation of ODE. Multiple tools use these approaches, but the focus will be on the two tools used throughout this work. Regardless of the approach employed, all simulations share the core features of obtaining a known GRN and generating data that fits it. Further, all simulations are simplified models of the biological system, attempting to mimic the system without capturing the true complexity of it. The reduced complexity comes with both advantages and disadvantages. On the one hand, it makes the system far more understandable, however, if the model is too simple, the simulation may not be informative. The two sides must be carefully balanced. While a more complex model may be more informative, it will also contain a larger set of parameters that must be optimised and tend to be computationally much more complex. It is impossible to simulate reality perfectly due to the high number of molecular steps and unknown effectors that have yet to be discovered. Determining how much of the real system is relevant to the topic that should be studied with the simulation is crucial. As a general rule, the simulation should only include the parameters necessary for the study at hand [61]. In GRN simulations, this has generally been limited to RNA and protein expression. Thus, the models capture only part of the biological GRNs, focusing primarily on transcription factors mediated regulation. As the transcription factor mediated regulation is the best-understood part of the regulatory system, it is advantageous to model only this part.
Figure 7: General workflow of a GRN simulation tool. The model consists of a GRN generation algorithm and a simulation model for data. Usually, both of these are based on properties observed in biological GRNs and experimental data. The network simulation either generates a novel network with a given topology or sub-selects part of a known biological GRN. The data simulation meanwhile uses mathematical models such as ODE or Bayesian models to simulate the dynamics described in the GRN. Most simulation tools also have functions to benchmark predictions performed on the generated network and data.
4.1 Modelling gene expression through differential equations

Today, the most commonly used simulation model is built around an ODE model originally developed by Ackers et al 1982 [62], utilising a hill function to model enzyme kinetic-like activation and repression of genes through transcription-factor mediated regulation. The model has been developed over many years [63], with the most commonly used implementation being the GeneNetWeaver tool released in 2011 by T. Shafftner et al [11]. It focuses heavily on capturing the dynamics of transcription factor-mediated gene regulation. The model consists of mRNA and protein expression as a joint system where the mRNA concentration is dependent on the protein concentration of the regulator, and the protein is linearly dependent on the mRNA expression of the corresponding gene. The change in mRNA concentration (equation 3) for a given gene \( x_i \) is described as the addition of new molecules, calculated as the max expression \( m_i \) times the activation level given by the function \( f(y_{reg}) \), minus the degradation of existing molecules, calculated as the degradation constant \( \lambda_{mRNA} \) times the current mRNA concentration \( x_i \).

\[
\frac{dx_i}{dt} = m_i \cdot f(y_{reg}) - \lambda_{RNA} \cdot x_i
\]

\[
\frac{dy_i}{dt} = r_i \cdot x_i - \lambda_{prot} \cdot x_i
\]  

(3)

Where \( m_i \) is the maximum transcription rate of the mRNA, determining how efficiently a gene is transcribed regardless of activation level, in terms of biology, it can be seen as a form of copy number representation. Ensuring both that activation cannot cause infinite transcription and allowing perturbation to be introduced to the system. \( \lambda_{i,mRNA} \) is the degradation coefficient for the gene \( i \), which determines how quickly the mRNA will degrade. \( r_i \) is the translation rate, and \( \lambda_{prot} \) is the degradation coefficient for the protein. The function \( f(y_{reg}) \) in equation 1 is a hill function describing the activation of the gene as a sigmoid relationship between the regulator and the target. As described by the equate:

\[
f(y_{reg}) = \frac{\alpha_0 + \alpha_1 \left( \frac{y_{reg}}{k} \right)^n}{1 + \left( \frac{y_{reg}}{k} \right)^n}
\]

(4)

Where \( \alpha_0 \) represents the passive expression of the gene without any regulation, \( \alpha_0 \) is often very small or zero and can then be removed from the equation entirely. \( \alpha_1 \) is a constant that determines how strong an effect the regulator has on the target. Both \( \alpha_0 \) and \( \alpha_1 \) correspond to relative expression and are
therefore between 0 and 1. $k$ is the dissociation constant which describes how strongly the regulator binds the target and thus the probability of a regulatory molecule being bound to the target gene. At $k=y$, the activation is 50% of the total possible activation. The model describes the interaction between the regulator and target as a probability function where the probability describes the likelihood of the regulator binding to the target and causing a transcription event. $n$ is the hill coefficient, which describes how quickly the transcription increases as the activation increases by controlling the steepness of the sigmoid curve.

Many models expand on these equations to incorporate regulatory behaviour requiring more than one regulator gene to affect the target. Collaboration between regulators is often done using an additive model where the effect of each regulator and regulatory combination are summed up (equation 5). It relies on the same variables as the single regulator (equations 3 and 4) but adds $P$ as a constant for how strongly the regulators work together, usually as a percentage value. It contains an $\alpha$ value for each regulator to allow for different levels of regulation from all regulators, both as combinations and single regulators.

\[
f(y_1, y_2) = \frac{\alpha_0 + \alpha_1 V_1 + \alpha_2 V_2 + \alpha_3 P V_1 V_2}{1 + V_1 + V_2 + P V_1 V_2}
\]

(5)

\[
V_i = \left( \frac{y_i}{k_i} \right)^n
\]

In equation 5, an example with only 2 regulators is shown, but in principle, this equation can be expanded to include any number of regulators with a combination for all regulators added. For three regulators, a term for $V_1V_2$, $V_1V_3$, $V_2V_3$ and $V_1V_2V_3$ would be added and so on for each added regulator.

The ODE model builds on several assumptions. A fundamental assumption is that the transcription and translation time is significantly higher than the time for other parts of the process. This assumption is based on previous studies showing that the half-life of both mRNA and protein molecules is higher than transcription, translation and transportation time within the cell [24,64]. Equally important is the assumption that a sigmoid hill-function can describe each regulatory interaction. This assumption is based on observations in biochemistry on how molecules in a solution, particularly enzymatic relationships, tend to follow a concentration-based sigmoid function that grows rapidly as concentration increases and then plateaus as all available substrates are already bound to the enzyme. For the model, this also means that there is a maximum expression, after which adding more activation will not cause any increase in expression. In biology, gene expression is limited by the copy number of a given gene, the binding kinetics between the regulator and target and
the total number of available transcription complexes. Other than this limitation, the model assumes for a cell where all other molecular complexes associated with gene and protein expression exist in excess of what is needed. All components needed for transcription and translation-related processes, such as RNA polymerases, ribosomes, transportation molecules, and nucleotides, exist in an infinite supply. Further, it assumes that degradation systems for both RNA and protein exist in concentrations greater than that of either molecule. However, the constants for transcript, translation and degradation can be modelled in a per-gene manner. Setting each constant separately allows for some variation between genes in how quickly they can be expressed and degraded. From a biological perspective, these differences can reflect things like the length of the gene, with longer genes requiring more time to transcribe and translate.

4.2 Using linear approximation to reduce model complexity

While the ODE-based model works very well to capture the dynamics of GRN, it is inherently a relatively slow algorithm as the model must be explicitly solved for each gene in the system. Suppose the dynamics of the system are not needed. For small systems around a steady-state, the model can then be simplified to a linear ODE system where the RNA expression of the target is directly dependent on the RNA expression of the regulator. The change in gene expression for GRN can thus be described as:

\[
\frac{dx}{dt} = Ax(t) + B(p(t) - f(t))
\]

\[
y(t) = Cx(t) + e(t)
\]

Where \(x \triangleq [x_1, x_2, \ldots, x_n]\) is a vector with the RNA expression, \(A\) is the interaction matrix which represents the interactions present in the GRN, \(p \triangleq [p_1, p_2, \ldots, p_n]\) is a vector containing all perturbations and \(f \triangleq [f_1, f_2, \ldots, f_n]\) is a vector containing potential perturbation noise. \(B\) is a matrix that describes the relation between perturbation and the state derivatives as the time point \(t\) changes. The response in the system is then described by \(y\) and depends on the matrix \(C\), which contains the relationship between the response and state and an unknown error \(e\). Working on this idea but focusing on capturing the change specifically between the two steady-states, a state where the change \(dx/dt = 0\), before and after perturbation T. Nordling [42] demonstrated that the system can be described by the linear equation:
\[ Y = A^{-1}(P - F) + E \]  

(7)

Which captures the change in gene expression based on the perturbation \( P \) and the interactions described in the interaction matrix \( A \) between the initial steady-state and the steady-state reached after perturbation. \( E \) and \( F \) are both matrices containing an unknown noise. Building on this model, a GRN simulation method called GeneSPIDER was developed by Tjärnber et al. in 2017. The model works under the assumption that each gene in the system is perturbed at least once, meaning that the size of \( X \) and \( P \) are identical. Under this assumption, the relative gain matrix, the inverse of \( A \), and the perturbation will describe the change \( X \). The model is much faster than ODE-based models, as only a single equation needs to be solved. Despite the simplification, the model can, under sufficiently small deviations from the steady-state, accurately capture the dynamics of a GRN.
5 Gene regulatory network inference

As discussed throughout this thesis inferring a GRN is a significant challenge, especially as the number of genes in the system grows larger. Due to this, numerous GRNI methods have been developed utilising various techniques to reconstruct the GRN based on gene expression. Here the focus will be on the methods used and developed throughout this work. For a more thorough review of the topic, see Mercatelli et al 2020 [65]. The methods used in this work primarily use single target perturbation, relying on a known perturbation scheme to find regulations in steady-state data. A brief introduction to other types of methods will be provided, but no details will be given as there exist a vast number of methods, and they are not relevant to the work at hand.

GRN methods presented here can be divided into six categories. First, the methods can be divided broadly into two categories based on which data type is used for the inference, either time series data or steady-state. As the name suggests, time series data is based on measurements taken over time, repeatedly measuring gene expression at set time points after the perturbation to obtain data on the continuous effect of the perturbation. This type of data has the advantage that more details can be captured of the GRN dynamics in the data as the continuous change should be captured across the time series. The downside is that the cost of generating this type of data for large data sets can grow exponentially as each experiment must be measured at each time point. In comparison, the steady-state method focuses on inferring a GRN from only two time points, one before the perturbation and one after the system reaches a new steady-state. A steady-state here refers to a point when the gene expression only exhibits small changes for a predetermined time, usually at least a few hours. This type of data is cheaper to generate, as fewer experiments are needed, but instead risks missing complex dynamics in the GRN as it is inferred from a snapshot of the system. Both types of methods have clear advantages over the other, and the methods of choice should be informed based on the current investigation. With a clear advantage of using steady-state data when creating initial GRN models focused only on where interactions exist. Time-series data, instead, offers significant benefits for capturing the underlying dynamics of the GRN [66].

Secondly, the type of perturbation a method works with allows it to capture different things in the GRN. Methods either work with single or multiple genes being perturbed. Regardless of what type of data or perturbation is used, all methods have one thing in common, something needs to be changed in the system in order to find any regulations. In theory, to optimally capture an entire GRN, all genes should be perturbed in all possible combinations. However, this type of experiment is not plausible as a nearly infinite number of experiments would be required for each species and possibly cell type. For
example, performing only the pairwise perturbations of all genes in a comparatively small genome, such as that of *E. coli*, which contains around 4225 genes [67], would require about 8.9 million separate experiments. Instead, research focuses on either single gene perturbation experiments, which focus on perturbing one gene after another and measuring the systemic change, or multi-perturbation experiments, where several genes are targeted simultaneously. Multi-perturbation experiments either target genes in specific combinations based on prior knowledge or more undetermined sets across the genome, usually based on a shared function [27].

Single gene perturbation experiments are generally performed using targeted molecules, such as RNAi experiments or CRISPR, which will, under optimal conditions, only affect the targeted gene. Single target perturbation experiment offers greater insight into the direct relation between a single regulator and its targets. As such, GRNI methods focusing on single perturbation experiments tend to excel at capturing novel interactions in the GRN. The downside of single perturbation experiments is that the data does not allow for the identification of complex regulation involving two or more genes. While the fact that multiple genes regulate a target gene will be captured, any collaborative effect cannot be captured. Further, as the experiments depend on changing a single gene at a time, the number of experiments needed to capture large systems often grows with the number of genes in the system. For this reason, single perturbation-based GRNI may be more suitable to identify novel regulatory relationships in smaller sub-networks and create initial topological GRNs. A topological GRN is a GRN that focuses on if an interaction exists but not so much on the strength or dynamics around interactions [66].

The multi-perturbation experiments cover a much broader set of perturbations, including topics like drugs, complex molecules or environmental perturbations, e.g. heat or glucose availability [27]. As the category is much broader, it is harder to define specific advantages for GRNI methods using it, as the advantages will vary with the nature of the perturbation. Experiments where genes have been targeted in specific combinations, for example, offer a deeper insight into how the perturbed genes interact with each other as they affect their regulation [68]. The data can thus, for example, be used to capture the previously described FFL interactions within the GRN, structures that, for single perturbation data, would be undetectable. Undetermined multi-perturbations further broaden the category of what the data can be used for. In general, this type of data can be used to capture novel interactions [69] in a similar way to the single perturbation data or to study functional effects in the GRN by targeting a specific function in the cell. The advantage of using multiple perturbation data for finding novel interactions is that the cost of these experiments will grow much slower and that, while less clear than specific combinations, combinatory regulations can still be captured. It, however, comes at the cost of the clearer perturbation and effect seen in single perturbation data.
Finally, methods can be divided into categories based on if they make use of the experimental design, using known perturbations to explain changes, or not. This division is probably the smallest with only a limited number of methods that relies on the perturbation information available [29]. The majority of methods instead opt to focus on identifying regulatory relationships using only the change in gene expression without any additional data.

5.1 Selecting the right GRNI method

With six different categories of methods to choose from for GRNI, the obvious question becomes which type of method is the best. As mentioned, when introducing each category, each type of method excels at certain tasks while not being able to handle others [70]. With that in mind, there is today no clear best choice of method, instead the GRNI method should be selected based on the available data and the goal of the study [71]. If no GRN exists for the genes of interest, single-perturbation steady-state data can, with advantage, be used to capture a detailed topology of the interactions. On the other hand, if the interest is to capture details of the dynamic interactions within the system, combinatorial perturbations in time-series data may offer the greatest insight into the GRN. The one exception to this uncertainty is when it comes to perturbation-based and non-perturbation-based methods, where the advantages and disadvantages are more well-known. Paper I shows that the perturbation-based methods can, on average, achieve higher correctness [29]. While this could be seen as a strong indicator for only using this type of method, they tend to come with stricter limitations compared to the non-perturbation-based methods. As methods using perturbation as input information relies on precise knowledge of the change introduced to the system, the available data is limited. Data generated for purposes other than GRNI is often unsuitable to use with these types of methods as often the perturbation is not as precise as they need it to be. In comparison, many non-perturbation-based methods can be applied to data with more general perturbations, e.g., metabolic or drug studies. In addition, most perturbation-based methods fall into the category of single-perturbation methods, meaning that each gene in the system needs to be perturbed one at a time. For most of these methods, that leads to the number of experiments growing directly with the number of genes studied, somewhat limiting the applicability when studying larger systems. In comparison, the non-perturbation-based methods usually depend only on gene expression across multiple unspecified perturbations, and adding non-perturbed genes does not affect their predictions negatively. By allowing for unperturbed genes to be included in the GRN, multiple-perturbation-based methods tend to be more suitable for larger systems. However, it may come at the cost of overall correctness in the predicted GRN.
5.2 Utilising least squares in GRNI

Most of the methods used throughout this work build on linear approximation described during section 4.2 Using linear approximation to reduce model complexity section. Utilising the same assumptions as the linear approximation model to capture gene regulatory interactions using least squares regression. Under the assumption that the gene expression is measured at steady-state this allows for the GRN to be captured through linearly fitting the perturbation to the observed change in gene expression between the control and the experiment.

5.2.1 Least squares with cutoff

The simplest method for capturing a GRN with least squares is the least squares with cutoff (LSCO) method, which does this directly by performing an ordinary least square fit between the relative change in gene expression and the perturbation. Using the change in the system to explain the effect of the perturbation through the GRN. In other words, by knowing the perturbation and observing any change following it, LSCO finds the GRN that best explains the effect of the perturbation. The GRN is thus found by fitting the relative change between the two steady-states to the known perturbation, with the goal of finding the GRN $A$ based on the perturbation $P$ and the relative change $Y$, usually given in log2 fold change:

$$A = -PY^{-1}$$  \hspace{1cm} (8)

$Y$ is, in general, a full matrix, a matrix that contains a non-zero value in most, if not all, cells. As such, the inversion $Y^{-1}$ will result in an $A$ matrix that contains a non-zero value in all cells. Note that unlike in equation 8 the noise matrices $E$ and $F$ are here unknown and assumed to be included in the $Y$ and $P$ matrix respectively. The values in $A$ correspond to the relative strength the regulation has. The values in $A$ will roughly correspond to how well they explain the perturbation, so a higher absolute value indicates a stronger regulatory effect from the perturbed gene on the changed gene. Using this fact, the LSCO algorithm relies on a magnitude-based cutoff where the absolute values closest to zero are gradually zeroed to remove false positive edges. As this is done in steps, the LSCO algorithm will return several networks where edges are gradually removed until all edges have been removed. It is up to the user to determine which predicted network they believe is the most accurate. A general rule that can be used to limit possible networks is that most biological GRNs tend to have between two to three edges per node [9].
5.2.2 Least squares with cutoff normalised

A central problem with LSCO relying on ordinary least squares is that this makes the method very sensitive to outliers and extreme values in the data. LSCO is particularly vulnerable to extreme value issues as it works with relative change after a perturbation. Relative change has an inherent issue: genes that are either lowly expressed or do not change much across the experiments will show a minimal relative change compared to all other genes. For example, a gene not regulated by any of the perturbed genes will consistently show a relative change very close to zero, as it will only change when directly perturbed in that system. Alternatively, if the gene is lowly expressed, the expression will likely be underestimated when measured before and after perturbation, caused by an innate issue with measuring the existence of infrequent RNA molecules. If the expression is underestimated, the change in expression becomes close to zero, with the minor change detected not caused by the perturbation but rather an issue with the method for measuring the expression. This results in all changes for the gene being close to but not equal to zero, and when the least squares fit is applied in LSCO, the regulatory effect is massively overestimated. The overestimation comes from the matrix inversion during the ordinary least squares fit, where values close to but not equal to zero become significantly higher than those values further away from zero. When combined with LSCO’s gradual removal of edges based on absolute values, these larger values will be conserved over any other value in the prediction. The conservation of these edges will lead to the overestimated gene being the only one with edges, and it will appear as if it regulates most of the genes in the system. As this comes from a mathematical error, there is no biological relevance to find, and the entire GRN model is false. Since the LSCO model has several properties that make it highly attractive as a GRNI method, a new method that expands on LSCO by adding a safeguard against overfitting was developed.

The method, Least squares with cutoff normalised (LSCON), was developed in paper II [72] and effectively removes the tendency of the original LSCO to overfit data. The method essentially works identically to the LSCO algorithm, starting with fitting the perturbation and relative change using ordinary least squares. LSCON then performs a column-wise normalisation according to the equation:

\[ \tilde{A}_{ij} = \frac{A_{ij}}{\sum |A_j|} N \]  

(9)

Where \( \tilde{A} \) is the normalised representation of the GRN matrix \( A \), \( i \) is the row and \( j \) is the column, \( N \) is the total number of rows in \( A \). Ensuring that the scale in each column is in a similar range, reducing any extreme values to a scale where they will no longer outweigh any other value. Once the normalisation
is applied, LSCON performs the same stepwise zeroing of values as LSCO, starting at the absolute value closest to zero and moving to the largest value in the predicted GRN. The relatively small change allows LSCON to robustly handle data with many extreme values while maintaining the same high performance demonstrated by LSCO in paper I. We show in paper II that LSCON maintains this high performance regardless of the presence of extreme values or not while maintaining a virtually identical run time to the original LSCO algorithm.

<table>
<thead>
<tr>
<th></th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
<th>E4</th>
<th>E5</th>
<th>E6</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-1.0000</td>
<td>-2.0000</td>
<td>0.2000</td>
<td>0.0000</td>
<td>-1.8000</td>
<td>0.2000</td>
</tr>
<tr>
<td>G2</td>
<td>0.0005</td>
<td>0.0003</td>
<td>0.0007</td>
<td>0.0005</td>
<td>0.0004</td>
<td>0.0005</td>
</tr>
<tr>
<td>G3</td>
<td>2.0000</td>
<td>0.3000</td>
<td>5.0000</td>
<td>2.2000</td>
<td>0.5000</td>
<td>-6.0000</td>
</tr>
</tbody>
</table>

**Figure 8: Underlying concept of the LSCON algorithm.** The figure shows the underlying workflow of the LSCON method and the issue that arises from small changes in the measured gene expression. When the original $Y$ matrix contains rows that consist of only values significantly closer to zero than the rest of the matrix, these values will be erroneously increased compared to the other values when inverted. LSCON handles this by normalising the columns in $A$ so that all values are on a more similar scale, creating the normalised matrix $\bar{A}$.

### 5.2.3 Penalised least-squares regression methods

While the ordinary least squares algorithm used in both LSCO and LSCON has shown to be highly effective, all ordinary least squares methods are inherently sensitive to noise in the data. Even relatively limited noise can drastically affect the algorithm, something that has long been known [73]. Several penalised least-squares algorithms have been developed to work around the noise sensitivity in ordinary least squares. The penalisation can take on many forms, but in general, it builds around the idea that by adding additional criteria to the least squares equation, weaker coefficients in the prediction will be further reduced, forcing the algorithm to focus more on the parts that explain more of the fit. The goal of the penalisation is to reduce the number of false positives...
as they often cannot explain as much of the data as the true positives. Of course, no penalisation model can completely eliminate false positives, and if the noise gets too high, these methods will still fail to generate meaningful predictions. The two most common penalised regression models in GRNI, and the ones used throughout this work, are the least absolute shrinkage and selection operator (LASSO) [74] and Ridge regression [75]. They work similarly to each other but use different norms to penalise the regression, with slightly different goals in mind. LASSO utilises the L1-norm to optimise the fit between the two variables by solving:

$$\overline{A}(\lambda) = \arg \min_A ||AY + P||_2^2 + \lambda ||A||_{l_1}$$

(10)

Where $A$ is the regression coefficients and $\overline{A}$ the penalised coefficients for the given $\lambda$. $P$ is the perturbation matrix that explains the observed effect contained in $Y$. $\lambda$ is a nonnegative regularisation parameter that can either be set or optimised for. Normally a series of $\lambda$ values are used, solving the above problem repeatedly to find the $\lambda$ value that optimises the solution. The key feature of LASSO is that as the value of $\lambda$ increases the number of non-zero values in $\beta$ will decrease, making the solution increasingly sparse. In effect what LASSO does is perform a feature selection in which variables can explain most of the effect. LASSO works by forcing the absolute sum of all coefficients in the fit to be below a fixed value forcing weaker coefficients to zero. LASSO’s ability to select features offers a great advantage when it comes to GRNI, as often, a large number of the features in the data will not contribute to explaining the observed effect.

Ridge regression, in turn, relies on an L2 penalty to reduce overfitting, focusing heavily on solving problems with ill-posed data and multicollinearity in the data. Ridge regression solves a problem nearly identical to that of LASSO, with the small but crucial difference in the condition it is optimised under which uses the Euclidean (L2) norm. Ridge regression thus solves the problem:

$$\overline{A}(\lambda) = \arg \min_A ||AY + P||_2^2 + \lambda ||A||_2^2$$

(11)

Where $A$ is the regression coefficients and $\overline{A}$ the penalised coefficients for the given $\lambda$. $P$ is the perturbation matrix that explains the observed effect contained in $Y$. $\lambda$ is a nonnegative regularisation parameter that can either be set or optimised for. Normally a series of $\lambda$ values are used, solving the above problem repeatedly to find the $\lambda$ value that optimises the solution. Unlike LASSO, Ridge regression will not set values to zero but rather reduce the values towards zero. Ridge regression works by forcing the sum of the squared coefficients to be less than a fixed value, the fixed value being set by the user.
or optimised based on selected criteria. Something that will force smaller co-
efficients towards zero, reducing their relative importance in the GRN. As
gene expression data is often ill-posed, Ridge regression can, in theory, reduce
sensitivity to noise. However, this was not observed in practice when testing
the method in papers I and II.

5.3 Non-perturbation-based methods

So far, all methods discussed have two things in common: they use the pertur-
bation design to construct a GRN through some form of least squares. While
regression-based methods are very common in GRNI, a majority of methods
do, in fact, not utilise the perturbation in any significant way to find a GRN.
The non-perturbation-based group of methods is harder to define, though, as
it contains a wide variety of different algorithms and approaches to GRNI.
Unlike the perturbation-based methods, which all have the joint approach of
using the perturbations to explain the observed effect, the non-perturbation-
based methods have no common factor other than being GRNI methods. For
example, among the more popular methods are Genie3 and CLR [56].

Genie3 relies on decision trees to determine how much the expression of
other genes can explain a gene's expression, assigning regulatory relationships
to those genes that explain a large part of the other genes' expression [76].
CLR, in turn, uses context likelihood of relatedness, which is also where it
gets its name from, to determine mutual information between the expression
of genes, how much the expression of one gene depends on another, measured
in a non-linear space [77]. While both aim to find how much one gene regu-
lates another, the approach is noticeably different from an algorithmic and
computational standpoint. Paper I also showed that the overlap in predictions
between Genie3 and CLR tends to be fairly low compared to that of the per-
turbation-based methods. What the non-perturbation-based methods do have
in common is that they usually use only gene expression to find the GRN.
Often relying on capturing how much a given gene affects the genes around it
across multiple experiments rather than by a known change. In its simplest
form, this can be done through correlation. However, more advanced methods
are usually needed due to noise and many non-linear gene regulatory interac-
tions. With the large variety of non-perturbation-based methods, they tend to
work better or worse depending on the experimental setup.
6 Present investigations

6.1 Knowledge of the perturbation design is essential for accurate gene regulatory network inference (Paper I)

Today, there exists a large number of methods for gene regulatory network inference (GRNI). Most methods take a distinct approach to the inference, utilising diverse methods such as random forest, mutual information or linear regression. The performance of the different methods will vary based on the approach to GRNI and the problem at hand. This paper evaluates the effect of utilising the perturbation design as prior knowledge when inferring regulatory networks from targeted single gene perturbation data. Five GRNI methods that use the perturbation information in their algorithm and five methods that do not use perturbation information were compared based on how well they can reconstruct a known gene regulatory network. The methods were tested in simulated data based on two separate tools, geneSPIDER and GeneNetWeaver, which create data using two distinct approaches to ensure any finding was not an artefact of the simulation method. The comparison of the methods shows a general trend where methods using the perturbation information tend to significantly outperform those that do not use the perturbation design. The findings suggest that more focus should be placed on knowing and utilising the information contained in the experimental design when reconstructing gene regulatory networks.

Future perspective
The study suggests a clear advantage in utilising the information contained in the experimental design, at least when working with single perturbation steady-state data. To fully understand the effect, the study could be expanded to include more data types, such as multi-perturbation or time-series data. In addition, demonstrating the same effect in a larger-scale experimental dataset would offer a stronger indication of the importance of utilising all available knowledge when inferring gene regulatory networks. For the field at large this type of large scale targeted perturbations have become a lot more feasible than previously thanks to methods like CRISPR and genome wide RNA sequencing. As such it is important to account for and utilise this additional information that were here shown to give even relatively simple methods a clear advantage.
6.2 Fast and accurate gene regulatory network inference by normalised least squares regression (paper II)

When selecting a method for inferring gene regulatory networks, two criteria are of utmost importance: first, how correct are the predictions performed and second, how long will the method need to run to perform predictions. In this paper, a method that encapsulates both criteria are presented, the Least squares Cut-Off with Normalisation (LSCON) method. The methods build on the previously developed method Least Squares Cut-Off (LSCO) but focus on improving the method's robustness. LSCO and LSCON rely on ordinary least squares to infer gene regulatory networks, a fast approach to inference that has been repeatedly shown to produce accurate predictions. The network $A$ is thus given by:

$$ A = -PY^{-1} \quad (6.2.1) $$

Where $Y$ is a size $N \times M$ matrix containing the relative gene expression change, $N$ is the number of genes measured, and $M$ is the number of experiments and replicates. $P$ is a matrix of size $Y$ containing the information about the corresponding perturbation for each column in $Y$. LSCO, however, suffers from a tendency to overeat the importance of genes that shows minimal change, a change close to but not equal zero across most or all experiments in the data. The overfit takes the form of the unchanging gene being strongly connected to all other genes. LSCON solves this issue by implementing a simple normalisation step where each column in the predicted network is normalised against its absolute sum such that the normalised network $\bar{A}$ is given by:

$$ \bar{A}_{ij} = \frac{A_{ij}}{\sum |A_{j}|} N \quad (6.2.2) $$

Where $A$ is an $N \times N$ matrix containing the least squares predictions, $i$ and $j$ is a given row and column respectively. $N$ is the total number of genes in the data. The normalisation will reduce columns based on their relative absolute sum, ensuring that abnormal columns created by the initial overfit are removed. LSCON were shown to be able to recover accurate gene regulatory networks from data with and without the outlier genes while remaining highly scalable due to a low time complexity.
Future perspective
LSCON was developed as a replacement for the older LSCO method, and it was shown to perform either better or equal LSCO in all tested cases. LSCON should, in theory, maintain that performance in any data where LSCO is applicable due to the use of the same underlying solution when inferring gene regulatory interactions. The normalisation in LSCON does not change the relative relation in the least squares coefficients. The study highlights the general importance of avoiding overfitting and to study the data properties when selecting a suitable method.

6.3 Generation of Realistic Gene Regulatory Networks by Enriching for Feed-Forward Loops (paper III)

Biological gene regulatory networks have been shown to contain two topologically conserved features: an edge distribution that approximately follows a scale-free distribution and an enrichment in feed-forward loop motifs. The feed-forward loop motif is a three-node motif where the first node regulates the second and third, and the second node, in turn, regulates the third node. For this work, a clear enrichment of feed-forward loops was shown for E. coli, S.cerevisiae, M.Musculus and H.sapiens, corresponding with earlier work in the field. A motif-based preferential attachment algorithm, FFLatt, was developed to create realistic simulated gene regulatory networks. The algorithm works by seeding a set of nodes and edges in the network, 20 nodes by default, and then adding nodes and edges to the network based on a set of four rules. Rule 1 corresponds to a node and edge added through preferential attachment, where the new node is connected to the existing network based on the probability function:

\[ P(g) = \frac{k_g^\gamma}{\sum_{i=1}^n k_i^\gamma} \]  \hspace{1cm} (6.3.1)

Where \( P(g) \) is the probability that the new node will connect to the existing node \( g \), \( k_n \) and \( k_g \) denote node-degree connectivity, and \( \gamma \) is a constant that determines the shape of the out-degree distribution. Rule 2 adds one new node and two edges from existing nodes in the network; rule 3 adds two edges between three existing nodes, but no nodes; and rule 4 adds a single edge between two existing nodes that both interact with the same third node. Rules 2, 3, and 4 all form feed-forward motifs, ensuring the motif’s realistic enrichment is maintained in the generated network. Which rule is selected is based on a priorly defined probability that was previously optimised to mimic biological
networks. Using these rules, FFLatt was shown to create more biologically feasible networks, particularly when compared with the popular simulation tool GeneNetWeaver. FFLatt networks were further shown to remain as stable as the compared network models as the network grows, a crucial property that prevents infinite growth of any gene.

Future perspective
The FFLatt algorithm presents a novel method of ensuring that the topological properties of biological gene regulatory networks can be conserved when generating networks making it suitable as a base for simulation tools; work on this is presented in paper IV. Overall, the paper highlights an interesting question for the field at large in how well the existing tools for simulating gene regulatory network data actually can represent the true complexity of the biological GRN. With many tools relying on older solutions for network generation, solutions that will not represent novel findings in the field and thus may miss important structural characteristics of gene regulatory networks. As such a deeper study of key properties in biological gene regulatory networks from both a topological and dynamic properties would be of great value to the field. FFLatt could potentially play a role in this through its modular nature.

6.4 GeneSNAKE: a Python package for benchmarking and simulation of gene regulatory networks and expression data (paper IV)

Benchmarking methods for gene regulatory network inference today face a central challenge caused by the limited availability of high-quality gene expression data corresponding to a known and verified gene regulatory network. As an alternative, the gene regulatory network field relies on simulated data. Several methods have been developed for this type of simulation. However, most of them struggle with some combination of an un-intuitive user interface and limitations in the properties that can be simulated. The methods often only cater to a specific simulation focusing on one or two properties the original authors envisioned as the most important for inferring gene regulatory networks. In an attempt to remedy this, the GeneSNAKE (Generation and Simulation of Networks and datA pacKagE) package was developed. GeneSNAKE offers a user interface that should be familiar to anyone with experience working with Python, with functions to generate variables for fast and easy creation of simulated data. To maintain maximum flexibility, GeneSNAKE allows for almost all variables to be set or updated if the user should feel the generative
functions are not enough. The GeneSNAKE package utilises the FFLatt algorithm to create realistic gene regulatory networks as a basis for its simulations. While FFLatt comes pre-packaged with GeneSNAKE, it also contains functions that allow for the usage of almost any network as the basis for simulations. For simulations of gene expression dynamics, the package uses an ODE model that has been repeatedly shown to capture the effect of gene regulation well. Finally, to the authors' knowledge, GeneSNAKE is the first simulation tool that allows for a fully customisable perturbation of the system, both in terms of design and effect. GeneSNAKE data were validated using a set of methods that have previously been shown to accurately reconstruct gene regulatory networks from expression data. The method shows that the underlying gene regulatory network is reflected in the data.

**Future perspective**
GeneSNAKE still requires some tweaking to be considered correct, especially the noise models mentioned above. The package is designed to be modular, allowing for future additions and modifications to keep the GeneSNAKE package relevant as new gene regulatory network properties become known.
7 Discussion and concluding remarks

GRNs are complex systems that play key roles in many aspects of life, from the maintenance of cellular processes to the evolutionary effects of altered regulatory interactions. Despite the importance of the GRN the system is not yet fully explored, while the field have made significant strides in recent years [78] a lot of research remains. To that aim this work focusing on the development and, in particular, testing of new methods for inferring GRNs. Primarily the work presented here focuses on developing tools for simulation of GRN dependent gene expression data and the use of this data for evaluation of GRNI methods.

In Paper I work is presented showing that GRNI can be significantly improved by utilising the information contained in the experimental design by using known perturbations to explain the observed change in gene expression. The study presents a solid foundation to build upon with a clear establishment of how additional information can be used to improve GRNI with straightforward methods like ordinary least squares being able to outperform traditionally top performing methods. A clear limitation of the study here is the examined data, both in terms of limited size with only 100-genes being tested and the data being of a single experimental design, data generated from the difference between control and a single time point with each gene being perturbed on at a time. As such a broader benchmark building on the findings presented in paper I testing additional sources of prior information, and more varied data would be of immense value to the field. Other studies have shown that using priors such as knowledge of the transcription factors in the system can improve GRNI, something that for example the Genie3 tool [76] tested in paper I already supports. Similarly, Kamal et al 2023 [79] showed that the knowledge of enhancer regions and the epigenetic status of the genes can be leveraged to infer and evaluate GRNs. The results of paper I in mind and the increasing availability of additional genomic information available today suggests that an area of focus of GRNI development should be on methods that can leverage this prior information rather than focusing solely on the more traditional approach of looking only at the change of gene expression.

Paper II presents the novel algorithm LSCON, a method that improves on an existing well performing tool by increasing robustness. The study builds on the concept that least squares coefficients can be rescaled without changing the relative importance. LSCON uses this concept to reduce the effect of fitting outliers in the data in the solution. For the LSCON study the primary focus were to be able to include genes of interest that showed a consistently limited change expression. The study, however, draws important attention to the importance of knowing both the properties of the data and the method used for GRNI. Showing that under the conditions presented here the otherwise top
performing method LSCO completely failed in its performance due to a mathematical artifact. The study also indirectly shows the importance of broad and varied benchmarking with realistic properties from experimental data. The GRNI methods tested here being affected by the single modified data property, one or more unchanging genes present in the data, to various degrees.

Building on the previous findings of how important accurate and varied simulations are paper III presents a novel algorithm, FFLatt, for generating biologically feasible GRN like networks. The study shows that the developed algorithm generates networks similar to that of the *E. coli* GRN. Using the developed FFLatt algorithm paper IV aims at developing a GRN based gene expression simulation tool, GeneSNAKE, offering the capability to simulate a broad range of network and data properties. The tool offers generation of data both over time and between stead-states for a multitude of experimental designs and cellular conditions offering GRN researchers the possibility to generate highly specific benchmarks to study topics such as the effect certain types of perturbations, data properties or even network structures may have on GRNI.

In conclusion the work here shows the importance of comprehensive benchmarking in the GRN field as well as provides means for performing this. With an ever-increasing amount of method for GRNI [57,80] and a deeper understanding of underlying properties for gene regulation this need for comprehensive benchmarks will only increase. Necessitating collaborative efforts in the GRN field to create and maintain up to date benchmarking tools and datasets.
Populärvetenskaplig sammanfattning


Att studera GRN är dock inte trivialt, nätverket själv har flera komplexa samspeglade komponenter som kan vara svåra att detektera. Metoderna för att mäta effekterna är också ofta indirekta och kan inte direkt mäta den reglerande effekten utan bara ändringar i genuttryck. För att komma runt detta görs mycket av forskningen på GRN genom matematiska simulationer. Denna avhandling syftar till att undersöka och utöka dessa simuleringar baserat på egenskaper observerade i cellulära GRN, samt att utvärdera hur metoder för att återskapa GRN påverkas av dessa egenskaper.
Acknowledgment

Throughout my time as a PhD student, I have had the luck of having the support of several amazing friends and coworkers. A big thank you to all the members of the Sonnhammer group. Erik Sonnhammer for your continued support, encouragement, and advice. Daniel Morgan, Deniz Seçilmiş, Erik Zhivkoplias, and Mateusz Garbulowski for our work together and advice with the GRN research we have performed during the years. Davide Buzzao, Emma Persson, Miguel Castresana-Aguirre and, Dimitri Guala for your advice and outside perspective on the research. Without all of you this PhD would not have been possible. Finally, a thank you to all my friends and family who have supported me through the many years of academia.
References


42. NORDLING TEM. ROBUST INFERENCE OF GENE REGULATORY NETWORKS: SYSTEM PROPERTIES, VARIABLE SELECTION, SUBNETWORKS, AND DESIGN OF EXPERIMENTS. 2013.

43. AALTO A, VITASAAKI L, ILMONEN P, MOMBARTS L, GONCALVES J. GENE REGULATORY NETWORK INFERENCE FROM SPARSELY SAMPLED NOISY DATA. NAT COMMUN. 2020;11: 3493. doi:10.1038/s41467-020-17217-1


45. ALON U. NETWORK MOTIFS: THEORY AND EXPERIMENTAL APPROACHES. NAT REV GENET. 2007;8: 450–461. doi:10.1038/nrg2102


52. WELDMICHAEL T, ASEMOLOYE MD, MARCHISIO MA. FEEDFORWARD LOOPS: EVOLUTIONARY CONSERVED NETWORK MOTIFS REDESIGNED FOR SYNTHETIC BIOLOGY APPLICATIONS. APPL SCI. 2022;12: 8292. doi:10.3390/app12168292


