Inferring gene regulatory networks using transcriptional profiles as attractors and its application on the white-opaque switch in *Candida albicans*

By
RUIHAO LI

A dissertation submitted
in partial fulfillment of the
requirement for the degree of
Doctor of Philosophy
in
Quantitative and Systems Biology

Committee in charge:

Professor David Ardell, Chair
Professor Suzanne Sindi
Professor Clarissa Nobile
Professor Aaron Hernday, Advisor
2023

Inferring gene regulatory networks using transcriptional profiles as attractors and its application on the white-opaque switch in *Candida albicans*

The dissertation of Ruihao Li, titled "Inferring gene regulatory networks using transcriptional profiles as attractors and its application on the white-opaque switch in *Candida albicans*", is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____ Date _____

Professor David Ardell, Chair

_____ Date _____

Professor Suzanne Sindi

_____ Date _____

Professor Clarissa Nobile

_____ Date _____

Professor Aaron Hernday, Advisor

University of California, Merced

2023

Inferring gene regulatory networks using transcriptional profiles as attractors and its application on the white-opaque switch in *Candida albicans*

by

Li, Ruihao

Doctor of Philosophy in Quantitative and Systems Biology

University of California, Merced

2023

Dissertation directed by Professor Aaron Hernday

ABSTRACT

Candida albicans is the most common cause of life-threatening disseminated fungal infections. It is capable of undergoing phenotypic switching between two distinct cell types, named 'white' and 'opaque'. The white-opaque switch is controlled by a complex genetic regulatory network (GRN) that consists of transcriptional regulators (TRs). The advent of methodologies for profiling mRNA transcript levels and specific protein-DNA interactions at a genome-wide level has greatly expanded our ability to determine the structure and output of genetic regulatory networks, however uncovering the logic of how these networks function remains a challenging endeavor. The field of genetic regulatory network inference aims to meet this challenge by using computational modeling to derive the structure and logic of GRNs based on the experimental data provided by these genome-wide approaches. Boolean, probabilistic, ODE-based, and other models have been developed to infer GRNs. However, most existing models do not incorporate dynamic transcriptional data, since it has historically been less widely available in comparison to "static" transcriptional data.

In this work, a novel evolutionary algorithm-based ODE model that integrates kinetic data and considers "static" transcriptional profiles as attractors has been developed to infer GRN structure. The model performed well on both *in-silico* and real-life datasets, and it was able to predict unknown transcription profiles produced upon genetic perturbations of the *Candida albicans* white-opaque GRN. Those genetic perturbations can be engineered *in vivo* and the result can be utilized to either support or further refine the model. Therefore, the model allows for an iterative refinement strategy to decipher GRN and verify its reliability: the model facilitates GRN candidate selection for experimentation and the experimental result in turn provides validation or improvement for the model.

DEDICATION

I dedicate this dissertation to

My parents Xiuhong Pan & Guoqing Li, who are always by my side.
My grandparents Chengzhong Li & Xiufeng Chao, who I remember lovingly.
My colleagues, for assisting and inspiring me.
My friends, for supporting me physically and mentally.

Yours sincerely,
Ruihao Li

ACKNOWLEDGMENTS

We gratefully acknowledge computing time on the Multi-Environment Computer for Exploration and Discovery (MERCED) cluster at the University of California, Merced, which was funded by National Science Foundation (NSF) Grant ACI-1429783. We especially extend our appreciation to Dr. David Ardell who provided valuable advice on the development of our approach. We also thank all Hernday and Nobile lab members for providing insight on the project. This work was supported by the NSF, National Institutes of Health (NIH) National Institute of Allergy and Infectious Diseases (NIAID), and NIH National Institute of General Medical Sciences (NIGMS) awards IIS1814405 and MCB1715826 to R.A., R15AI37975 to A.D.H., and R35GM124594 to C.J.N., respectively. This work was also supported by the Kamangar family in the form of an endowed chair to C.J.N. The funders had no role in the study design, data collection and interpretation, or the decision to submit the work for publication.

RUIHAO LI

Curriculum Vitae

Quantitative and Systems Biology Graduate Program
School of Natural Sciences, University of California, Merced
(323) 449-4365
rli46@ucmerced.edu
Github

Education

2017–present **PhD, Quantitative and Systems Biology**, *University of California, Merced*, USA.
Genetic regulatory network inference, Mathematical modeling of biological systems, Genomic and proteomic sequence, Molecular biology, Microbiology, Machine Learning and Deep Learning

2016–2017 : **Exchange Student, Integrative Biology**, *University of California, Berkeley*, USA, Fully funded by studying abroad & international exchange student program for National Excellent Undergraduates, China Scholarship Council.

2013–2017 : **Bachelor of Science, General Biology**, *Huazhong University of Science & Technology (HUST)*, China.

Publications

In Progress

2023 Mohammad Qasim, **Li, Ruihao**, Morgan Quail, Clarrisa Nobile, and Aaron Hernday. Interplay between chromatin organizing transcription factors and dynamic nucleosomes regulate cell fate decisions and developmental processes in *Candida albicans*. ***Preparing for submission***, 2023.

Published Articles

2023 **Li, Ruihao**, Jordan C Rozum, Morgan M Quail, Mohammad N Qasim, Suzanne S Sindi, Clarissa J Nobile, Réka Albert, and Aaron D Hernday. Inferring gene regulatory networks using transcriptional profiles as dynamical attractors. ***PLOS Computational Biology***, volume 19(8), page e1010991. Public Library of Science San Francisco, CA USA, 2023.

2021 Mingyue Cheng, Zhangyu Cheng, Yiyan Yu, Wangjie Liu, **Li, Ruihao**, Zhenyi Guo, Jiyue Qin, Zhi Zeng, Lin Di, Yufeng Mo, et al. An engineered genetic circuit for lactose intolerance alleviation coupled with gut microbiota recovery. ***BMC biology***, volume 19(1):137. BioMed Central, 2021.

2016 Shuyan Tang, Wang Xi, Zhangyu Cheng, Lei Yin, **Li, Ruihao**, Guozhao Wu, Wangjie Liu, Junjie Xu, Shuaiying Xiang, Yanxiao Zheng, et al. A living eukaryotic autocementation kit from surface display of silica binding peptides on yarrowia lipolytica. ***ACS Synthetic Biology***, volume 5(12), pages 1466–1474. ACS Publications, 2016.

Presentations

2023 **Li, Ruihao**, Jordan Rozum, Morgan Quail, Mohammad Qasim, Suzanne Sindi, Clarrisa Nobile, Réka Albert, and Hernday Aaron. Inferring gene regulatory networks using transcriptional profiles as dynamical attractors. ***The*** 22^{nd} ***International Conference on Systems Biology (Talk)***, 2023.

2023 **Li, Ruihao**, Morgan Quail, Mohammad Qasim, Suzanne Sindi, Clarrisa Nobile, and Hernday Aaron. Inferring gene regulatory networks using transcriptional profiles as dynamical attractors and its applications. ***The Math Bio Seminar at University of California, Merced (Talk)***, 2023.

2019 **Li, Ruihao**, Morgan Quail, Mohammad Qasim, Suzanne Sindi, Clarrisa Nobile, and Hernday Aaron. Gene regulatory network reverse engineering using attractor-based evolutionary algorithm. ***The Math Bio Seminar at University of California, Merced (Talk)***, 2019.

Research Experience

University of California, Merced

2018 – present **_Inferring the Genetic Regulatory Network that Governs the White-opaque Switch in $Candida\ albicans$_**, *General purposes: fungal pathogenesis, antifungal therapies development*.

- Built an **ODE-based model implemented with logic gates** to simulate the biological processes of transcription, translation, and transcriptional regulation in a casual gene regulatory network.
- Wrote an **evolutionary algorithm** to automatically and iteratively reconstruct the ODEs according to the gene regulatory network structure, solve the ODEs numerically, select the networks whose ODEs can better reproduce the transcriptional profiles as steady states, and mutate the network structure.
- Ran the evolutionary algorithm on a Linux-based high-performance computing cluster and applied **parallel computing** to accelerate the speed of the algorithm.
- Demonstrated a **proof-of-principle** that illustrates how the network structures are linked to their steady states. Tested the performance of the approach on $in\text{-}silico$ datasets and made **comparisons** against six other leading methods.
- Applied the approach to the **white-opaque switch core circuit** in $Candida\ albicans$ and inferred a network. Made **testable predictions** on the transcriptional profile produced upon genetic perturbation.
- Tested the predictions experimentally with **genome-editing** tools and RNA sequencing and further refined the model.

Advisor : **Dr. Aaron Hernday**, *Full Professor, Department of Molecular Cell Biology, University of California, Merced*(*Personal Web-page*)

Collaborator : **Dr. Réka Albert**, *Distinguished Professor of Physics and Biology, Pennsylvania State University*(*Personal Web-page*)

2019 – present **_Identifying the Roles of Chromatin Accessibility in Regulating the White-opaque Switch in $Candida\ albicans$_**, *General purposes: epigenetic landscape characterization*.

- **Mapped, analyzed, and visualized** the ATAC-seq and MNase-seq data of white and opaque $Candida\ albicans$.
- Performed **nucleosome occupancy prediction** on $Candida\ albicans$ genome and compare the result against experimental data;
- Performed **intrinsic disorder region prediction** on $Candida\ albicans$ proteome.

Advisor : **Dr. Aaron Hernday**, *Full Professor, Department of Molecular Cell Biology, University of California, Merced*

2018 – 2019 **_Indentifying Novel Transcription Factors Involved in the White-opaque Switch in $Candida\ albicans$_**, *General purposes: transcriptional regulation study, antifungal therapies development*.

- Mapped the RNA-seq data of various TF knock-out mutants and performed **differential expression analysis**.
- Applied **logistic regression and prion-like amino acid composition scores** to select the potential transcription factors.
- Knocked out the potential transcription factors and measured the white-opaque switch frequencies.
- Applied **support vector machine** to classify white and opaque cells with different switching frequencies.

Advisor : **Dr. Aaron Hernday**, *Full Professor, Department of Molecular Cell Biology, University of California, Merced*

Huazhong University of Science & Technology (HUST)

2016 – 2018 ***Signal filter: A toolkit controlling gene expression based on negative feedback multi-stable state circuits***, *General purposes: lactose intolerance alleviation*.

- Independently responsible for research proposal and **design of the gene circuit**: a negative feedback tri-stable system with the capability of reducing noise and converting pulse signal into a robust and persistent signal.
- Performed gene circuit analysis followed by mathematical modeling.

Advisor : **Dr. Kang Ning**, *Associate Professor, College of Life Science & Technology*, Huazhong University of Science & Technology (HUST) (*Personal Web-page*)

2015 – 2017 ***Innovation in Technology of DNA-based Preservation and Access of Digital Information***, *General purposes: high-density storage of digital information*.

- Translated a poem between binary codes and DNA sequence using **Huffman coding**.
- Studied the **robustness and fidelity of DNA sequence** subject to changing pH, temperature, shear force, and salt concentration.

Advisor : **Dr. Kang Ning**, *Associate Professor, College of Life Science & Technology*, Huazhong University of Science & Technology (HUST)

Fellowships & Awards

2018 ***QSB Summer Research Fellowship*** of University of California, Merced.

2016 ***GOLD MEDAL***, INTERNATIONAL GENETICALLY ENGINEERED MACHINE (iGEM) COMPETITION, Boston, Massachusetts, USA.

2015 ***GOLD MEDAL***, INTERNATIONAL GENETICALLY ENGINEERED MACHINE (iGEM) COMPETITION, Boston, Massachusetts, USA.

General Research Skills

Machine Learning **Regression, Classification, Clustering, Dimensionality reduction, Neural nets, Deep learning, Big data, Graph model, etc.**.

Programming **Python, R, Matlab, C++, Perl, Shell/Bash, MySQL, HTML, etc.**.

Wet Lab Techniques **Genome editing technique, Molecular cloning, SDS-PAGE, ELISA, qPCR, etc.**.

Teaching Assistantship

Fall, 2017 : **BIO 002L: Molecular Biology Lab**.

Spring, 2018 – Spring, 2020: **BIO 120L: General Microbiology Lab**.

Fall, 2020 : **BIO 001: Contemporary Biology**.

Spring, 2021 : **BIO 140: Genetics**.

Fall, 2021 : **BIO 001: Contemporary Biology**.

Referees

Dr. Aaron Hernday
Associate Professor, Department of Molecular Cell Biology
University of California, Merced
☎ (209) 228-2450
✉ ahernday@ucmerced.edu

Dr. Clarissa Nobile
Full Professor, Department of Molecular Cell Biology
University of California, Merced
☎ (209) 228-2427
✉ cnobile@ucmerced.edu

Dr. Réka Albert
Distinguished Professor, Department of Biology and Physics
Pennsylvania State University, PA
☎ (814) 865-1141
✉ rza1@psu.edu

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

TABLE

LIST OF FIGURES

FIGURE

CHAPTER I

INTRODUCTION

1.1 A brief introduction to genetic regulatory network reverse engineering

It has been widely accepted that genetic regulatory networks (GRNs), which are comprised of interactions between sequence-specific DNA-binding proteins, or transcription factors (TFs), and their respective regulatory target genes ([1]), are one of the primary underlying mechanisms by which unique cells respond to environmental variables ([2]), maintain homeostasis ([3]), develop into multicellular organisms ([4]), and make cell fate decisions ([5]). Inferring the architecture of GRNs based on experimental datasets, also known as the "inverse problem" ([6]), is important to understanding these cellular processes (see [7, 8] for examples). The advent of high-throughput "omics" techniques ([9]) has dramatically accelerated the pace by which researchers can obtain these experimental datasets for GRN reverse engineering ([10]). The most commonly used high-throughput approach is RNA sequencing, which effectively and economically identifies and counts the number of transcripts present for each RNA species, and thus generates a transcription profile of the cell or tissue being assayed. With multiple bulk or single-cell transcription profiles measured at different time points, or in different cell types, one can see which genes have been up- or down-regulated and further infer the logic of the GRNs that underlie those regulatory changes ([11]).

1.2 Existing GRN inference approaches and challenges remained

In the past twenty years, numerous modeling approaches have been developed to infer GRN architectures using "omics" data and other prior knowledge ([12, 13, 9, 14]). GRN inference models can be broadly categorized into four distinct categories, based on

the algorithms and hypotheses they employ (see reviews: [15, 16, 17, 18, 19, 20, 21, 22, 23]).

1.2.1 Data-driven models

The models of this type do not simulate the biological processes such as transcription or translation, but hypothesize that interacting genes have correlated expression levels, which may result from common regulators and shared biological characteristics. The correlation in gene expression does not imply causality since indirect regulation common exists in biology and can give rise to correlated gene clusters. Specifically, these models apply the Pearson correlation, mutual information, and linear regression ([24]) to infer GRNs. Pearson correlation is calculated by the ratio between the covariance of two genes' expressions and the product of their standard deviations (Eq. I.1):

$$\mathbf{corr}(X, Y) = \frac{\mathbf{cov}(X, Y)}{\sigma_X \cdot \sigma_Y}, \tag{I.1}$$

where X and Y denotes the expressions of gene X and Y. $\mathbf{cov}(X, Y)$ is the covariance and the σ_X and σ_Y are the standard deviations. For instance, a method called MTPCC and built by [25] performs Hierarchical clustering using the Pearson correlation coefficients as the metric of distance. MTPCC uses a threshold to establish the boundaries among gene clusters and evaluates the significance of the relationships among the clusters.

Mutual information measures how much the certainty of a gene's expression can decrease the uncertainty of another gene's expression. It is calculated by Eq. I.2:

$$\mathbf{MI}(X, Y) = \sum_{x \in X} \sum_{y \in Y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)}, \tag{I.2}$$

where $\mathbf{MI}(X, Y)$ is the mutual information; $P(x, y)$ is the joint probability mass function of gene X and Y, and $P(x)$ and $P(y)$ are the marginal probability mass function for gene X and Y respectively. A widely used example is a method called ARACNE ([26]). Briefly, ARACNE obtains an adjacency matrix with MI values for each gene pair and removes the non-significant and indirect interactions by a metric known as data processing inequality. A limitation to the mutual information methods is that the estimation of the joint probability can be highly sensitive to noise when the sample size is small.

Linear regression models assume that a gene's expression can be explained by the linear combination of other genes' behaviors. Typically, a weight coefficient matrix is applied to represent how significantly other genes can impact the target gene (Eq. I.3).

$$X_{g,i} = \sum_{j \neq g} W_j X_{j,i} + \varepsilon_i, \tag{I.3}$$

where $X_{g,i}$ denotes the expression value of the g^{th} gene in the i^{th} sample. W_j is the weight coefficient matrix for the j^{th} gene and ε_i is the noise term. The weight coefficient matrix can contain numbers with both signs. Positive and negative numbers represent activating and inhibiting genes while zeros mean no regulation interaction. Normally, the weight coefficient matrix is trained iteratively in order to minimize the difference between the estimated gene expressions and the data. A method called KFLR, proposed by [27], implemented both mutual information and linear regression to construct a directed GRN and showed better performance compared with several well know methods. However, real biological systems often exhibit non-linear behavior in gene regulation and enzymatic catalysis due to saturation effect. A certain deviation may be introduced inevitably by linear regression.

These data-driven models are often simple because they normally do not require prior knowledge or expensive computation. They are powerful when dealing with large-scale GRNs having hundreds or thousands of genes. However, they cannot determine if a gene is a regulating gene or a target gene: their outcome GRNs are symmetric matrices. Additional downstream analysis can be performed to further determine the roles of the players within the GRN.

1.2.2 Probabilistic models

Probabilistic consider gene expression as a random process and infer GRN architecture by maximizing the likelihood of a GRN reproducing the input transcription profiles using heuristic algorithms ([28, 29]). Bayesian network approaches are an outstanding representative of probabilistic models. In a Bayesian network, nodes represent gene expression as random variables and edges are the conditional dependence. Since conditional dependence can indicate direction, the outcome Bayesian network is a directed graph, or an asymmetric adjacency matrix. Given a network structure, the joint probability distribution of all genes is calculated by Eq. I.4:

$$P(x_1, x_2, ..., x_n | A_{net}) = \prod_{i=1}^{n} P(x_i | TF_i), \tag{I.4}$$

where $P(x_i|TF_i)$ denotes the conditional probability distribution and TF_i are the transcription factors regulating gene i. Typically, an (local) optimal network structure is searched by maximizing the logarithm joint probability distribution. To guarantee that the joint probability distributions are normalized to 1, the Bayesian network has to be acyclic, which means a gene without any transcription factors must exist in the network. Bayesian network approach is popular in the field of GRN inference since it regards the expressions of genes as random variables and therefore can effectively deal with the noise within the data. Bayesian network approach is also flexible and can incorporate different types of data and prior knowledge. Many models have been developed using this approach, such as ScanBMA ([30]), AR1MA1-VBEM ([31]), BGL ([32]), and SCOUP ([33]). However, the Bayesian network approach is computationally expensive because searching for the optimal network structure is a NP-complete problem ([34]). As a result, it usually cannot deal with large-scale GRNs. In addition, self-regulating genes and feedback loops are important and widely exist in biological networks but the Bayesian network cannot capture these features since it does not allow for any loops.

Dynamical Bayesian network (DBN) is an extension of the Bayesian network and it has an additional assumption that the network is subdivided into a sequence of time steps, each containing the same number of genes ([35]). The expression of a gene at the current time point could only affect the expression of other genes, or itself, at the next time point. DBN can construct cyclic networks since a gene can regulate itself at the next time point. Limitations for DBN still exist: the time complexity for DBN is even higher than Bayesian network due to the implementation of time-series data. Moreover, There can be multiple valid factorizations of a joint probability distribution, leading to different networks encoding exactly the same probability distribution. This issue is known as Markov equivalence in probability theory. A common way to resolve this problem is to build an "essential graph" that can represent the equivalence class ([36]).

1.2.3 Boolean models

Boolean models are capable of simulating how GRNs control the expression of genes over time (i.e. GRN dynamics) without prior knowledge of any kinetic parameters since they use binary variables, 1 and 0, to define the state of a gene as "on" or "off" ([37]). Therefore, a GRN comprised of n genes has 2^n different states. A transition matrix, also known as a truth matrix built by a combination of logic gates such as AND, OR, and NOT, is used to decide how the system transits from one state to the next. Given an

initial state and a transition matrix, a trajectory of subsequent states can be computed. The recurring or steady states are referred to as attractors, towards which a kinetic system tends to evolve and converge. Boolean models have the advantage of simulating GRN dynamics without kinetic parameters and are able to apply attractor dynamics to infer GRNs ([38, 39]). Numerous Boolean models, such as BoolNet ([40]) and SimBoolNet ([41]), have been developed and successfully applied in a variety of biological systems ([42, 43, 44]). Kauffman ([45]) has originally suggested that stable cell types (represented by their transcriptional profiles) can act as attractors. This view provided a modeling basis for Waddington's epigenetic landscape ([46]) and has been supported by additional studies ([47, 48, 5, 2]). Additionally, it has been applied to GRN inference in Boolean models ([49, 50, 51, 52, 53]) and their result suggested that a biologically plausible GRN architecture can be identified by matching Boolean network attractors to different cell types that are characterized by their respective binary transcription profiles, and the output generated by the inferred GRNs was in agreement with the experimentally observed data. The application of attractor dynamics can compensate for "the curse of dimensionality" brought by the complexity of GRN architecture ([54, 55, 50]); when the scale of a GRN increases, the number of potential GRN architectures grows so fast that the available data points become sparse in the state space (Fig. 1.1a and 1.1b), and the data sparsity can undermine the significance and robustness of the model results. By employing binary transcription profiles as attractors, the Boolean models can compensate the curse of dimensionality since the data points become "basins of attraction", which can cover significantly more state space and provide more information on the system dynamics (Fig. 1.1c). Although the simplicity of Boolean model allows the beneficial application of attractor dynamics, it limits its faithfulness to real-life GRN system, where gene expression levels (commonly represented by the number of transcripts for each RNA species) are most accurately described as a continuous variable rather than binary "on" vs. "off" states ([56, 57]).

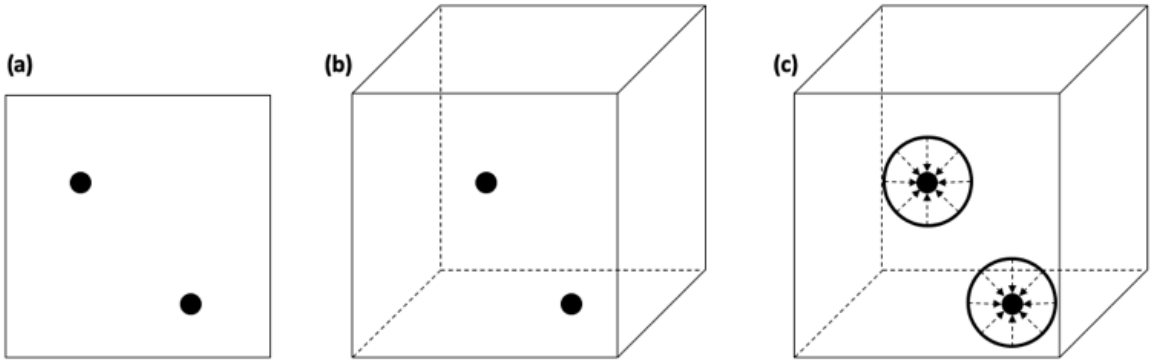


Figure 1.1: (a) (b) Curse of dimensionality: when dimensionality increases (from 2 to 3), the state space volume grows dramatically, and the existing data points become sparse passively. (c) When the data points become attractors, their basins of attraction can cover more space and compensate the curse of dimensionality.

1.2.4 Ordinary Differential/Difference equation (ODE) models

ODE-based models can simulate the dynamics of gene expression processes in a quantitative manner based a set of ordinary differential/difference equations. Specifically, the ordinary differential equations are established to describe how each gene is regulated by other genes and how the gene expressions change over time (I.5).

$$\frac{dx_i}{dt} = f_i(TF_i, \theta, u_i), \tag{I.5}$$

where $f_i(TF_i, \theta, u_i)$ denotes the regulation function for the i^{th} gene and u_i is the external perturbation added to the gene. The regulation function can be linear, such as a weighted average of other genes' expressions, or non-linear, such as a sigmoid function or Hill function (see examples: [58, 59, 60, 61, 62, 63, 64, 65]). Differential equations can be solved numerically or analytically. Numerical methods use finite numbers to compute the integrals defined by the ODEs step by step, while analytical methods, also known as symbolic computation, use mathematical symbols to calculate the solutions directly. Therefore, an analytical solution is perfect because it has absolutely no error. However, it has been proved that complex ODEs, such as Eq. I.5, cannot be solved analytically (i.e. in closed-form expression) ([66]). Specifically, if one wants to write down the analytical solution of the ODEs, if it exists, with normal operations and functions, such as plus, minus, multiplication, exponent, logarithm, and trigonometric functions, infinite amount of ink is needed. In fact, analytical solution is not practical for most complex non-linear differential equations, and numerical solution is the only option to solve them. Therefore,

ODE models are the most computationally expensive approach of all and commonly limited to small-scale GRNs. On the other hand, ODE models can incorporate any types of data and prior knowledge to faithfully represent the biological processes, such as transcription and translation. The outcome of ODE models can provide detailed structural information on the GRN including the direction and sign of the edges.

Since ODE models can quantitatively simulate GRN dynamics with discrete or continuous gene expression levels, they are related to Boolean models ([67, 68, 69, 70, 71]) and also have the potential to apply attractor dynamics ([72, 73]). For instance, FOS-GRN ([74]) and Netland ([75]) can reconstruct multi-attractor kinetic landscapes with ODEs and user-defined parameters. In addition, Drs Jin, Liu, Mochizuki, Rozum, Wang, Zanudo and their colleagues ([76, 77, 78, 79, 80, 81, 82]) have studied the controllability of GRNs (whose topologies are known) and demonstrated how the GRNs can be steered from one attractor state to another using ODE-based models. Typically, existing approaches to infer GRNs in ODE-based models fall into two categories: gradient matching and trajectory matching from time-series gene expression data ([83]). Gradient-matching approaches do not solve the ODEs but directly compute the gradient of gene expression data using Gaussian process regression and then optimize the parameters of the ODE system ([84, 85]). On the other hand, trajectory matching aims at tuning parameters to minimize the discrepancy between a computed trajectory and the corresponding observed one ([86, 83]). The attractor-matching approach, which does not focus on specific trajectories or gradients, has not been applied due to the lack of experimentally measured GRN kinetic data ([56, 22]), such as the rates of transcription and mRNA degradation, which shape the kinetic landscape, or vector field, of a GRN system and are essential for determining the attractors' positions and "basins" ([87, 74]).

There are no absolute boundaries amongst these categories and not all GRN inference models can be assigned to them ([88, 89, 90, 91, 92, 93, 94]). Nevertheless, the categorization provides a board view of existing GRN inference models and their pros and cons. According to the features of a biological system, a suitable GRN inference model can be chosen based on its speed of analysis, amount of data needed, faithfulness of biological reality, and the ability to perform prediction.

1.3 Challenges

Although transcription profiles are informative and have been widely used, it does not directly reflect whether a gene is activated or repressed by its regulators ([95]), since some

mRNAs are highly stable and can accumulate in cells, while others are actively degraded. Transcription rate should be the fundamental criterion to determine how actively a gene is being transcribed ([96]). Despite the importance of transcription rates in understanding GRN dynamics ([97, 98]), they were much less available than expression-based data in most organisms. For example, the yeast *Saccharomyces cerevisiae* is a well-studied model eukaryote, but its mRNA transcription and degradation rates could not be measured without cellular perturbation until 2011 when Miller et al. developed a method called "dynamic transcriptome analysis", which enables the acquisition of genome-wide transcriptional kinetics by non-perturbing RNA labelling with the nucleoside analog 4-thiouridine ([99]). More systematic and high-throughput measurement assays, including BRIC-seq ([100]), 4sU-seq ([101]), TT-seq ([102]), TUC-seq ([103]), SLAM-seq ([104]), TimeLapse-seq ([105]), and csRNA-seq ([106]), were invented even later (see Table 1.1). In addition, incorporating these kinetic parameters in GRN inference models can be difficult because they are also cell-type specific and often vary in different cellular conditions, such as growth stages ([107, 96]). As a result, most ODE-based models did not have access to the measured kinetic parameters and tried to estimate them while inferring GRN ([108, 109, 110, 111, 112]). This strategy largely varies from the application of attractor dynamics in which measured kinetic parameters are already known and are used to find and match the attractors.

Table 1.1: Description of the methods for mRNA kinetics measurement

Method	**Description**	**Reference**
BRIC-seq	BRIC-seq uses Bromo-urdine (BrU) to label endogenous mRNA transcripts, which are later pulled down by BrU antibody. The half-life of each transcript is calculated from the difference of labeled and unlabeled transcripts.	[100]
4sU-seq	4sU-seq uses 4-thiouridine (4sU) to label endogenous mRNA transcripts. 4sU-labeled RNAs are recovered by dithiothreitol, a reducing agent. The mRNA half-life is calculated from the ratio of 4sU-labeled and total RNAs by assuming steady-state conditions.	[101]

Continued on next page

Table 1.1 – continued from previous page

Method	Description	Reference
TT-seq	TT-seq includes a RNA fragmentation step before purification of 4sU-containing RNAs. This fragmentation allows the measurement of only newly transcribed 4sU-labeled RNA.	[102]
TUC-seq	In order to measure 4sU-labeled RNAs without biochemical enrichment, TUC-seq uses osmium tetroxide-mediated oxidation to convert 4sU into cytosine. By identifying the 4sU-to-C mutations in RNA sequencing, the initial 4sU-labeled RNAs can be recognized, and the mRNA half-life can be calculated.	[103]
SLAM-seq	SLAM-seq is similar to TUC-seq. The main difference is that SLAM-seq uses a thiol-reactive compound iodoacetamide to convert 4sU into cytosine.	[104]
TimeLapse-seq	TimeLapse-seq is similar to TUC-seq and SLAM-seq. The main difference is that TimeLapse-seq uses oxidative-nucleophilic-aromatic substitution to convert 4sU into cytosine.	[105]
csRNA-seq	csRNA-seq enriches short ($\sim$20-60 nt) and 5'-capped RNAs using denaturing gel electrophoresis and enzymes (RNase, phosphatase, and pyrophosphohydrolase). By measuring these specific RNAs, csRNA-seq can directly reveal the activity of transcription for each gene.	[106]

Another classic challenge in GRN inference is that while a wealth of high-throughput "omics" datasets are publicly available, or can be readily obtained through additional experimentation, it is still extremely challenging to directly determine the complete and comprehensive

composition and structure of "real-world" GRNs in living organisms ([113, 17]). Therefore, the use of experimental data in GRN inference can be problematic when it comes to validating the outcome of GRN model predictions, since one can rarely if ever be certain that the experimental data provides a complete picture of the real-world GRN structure. For this reason, it has become common practice in the field of GRN inference to utilize *in silico* (i.e. computer generated) datasets, which can provide gene expression data that is directly predicted based on a hypothetical "source" GRN model ([114, 115, 116, 117]). Essentially, GRN inference models are typically judged based on their ability to infer a hypothetical "source" GRN, which in turn has been used to generate the *in silico* datasets upon which the GRN model has been deployed. This seemingly circular approach has a distinct advantage over the use of experimental datasets, in that "true structure" of the "source" GRN is known. However, this approach also suffers from necessary simplifications and the lack of true biological complexity. In addition, while some *in silico* generators ([118]) allow user-given parameters, most others ([116, 114, 115, 119, 120]) do not provide their randomly sampled kinetic parameters along with their expression datasets.

These challenges call for a novel GRN inference model that can extend the application of attractor-matching strategy from Boolean model to ODE-based model by incorporating measurements of mRNA synthesis and degradation rates. For one thing, the newly developed genome-wide transcriptional kinetics assays and data can compensate the essential information missed in "static" RNA-seq data. For another, the ODE-based model can make full use of its advantages and provide detailed insights and predictions on the GRN structure and dynamics.

1.4 A simple testing GRN model

We intended to use a GRN in *Candida albicans* as a simple model system for testing our GRN inference approach. *Candida albicans* can switch between white and opaque cell types (Fig. 1.2), which significantly differ in virulence characteristics, metabolic preferences, mating ability, and morphology ([121]). Each cell type can be stably and heritably maintained for hundreds of generations, and yet switching between the two cell types occurs stochastically and spontaneously about once out of a thousand cell divisions in standard lab conditions ([122], [123]). In spite of the differentiation of the white and opaque cells, the primary sequence of the genome doesn't change and therefore this switch fits the classical definition of an epigenetic switch. Environmental conditions such as temperature, pH, and carbon source are able to affect the white-opaque switch. For

example, shifting the temperature from 25°C to 37°C could trigger the opaque-to-white switch ([122], [123]).

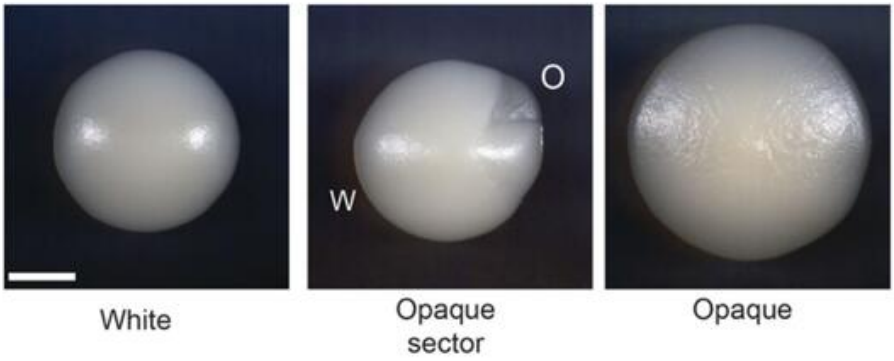


Figure 1.2: Cell and colony morphology of white and opaque *Candida albicans* and the white-to-opaque sectoring. Scale bar indicates 1 mm. Figure modified from [124]

.

The white-opaque switch is controlled by a complex genetic regulatory network (GRN) that consists of eight "core" transcriptional regulators (TRs). White opaque regulator1 (WOR1) is the master regulator of the opaque state ([125]). Wor1 expression is elevated in opaque cells, relative to white cells, and is essential for the transition to, and heritable maintenance of, the opaque cell type. The core switch has been extensively characterized by numerous genomic techniques including Chromatin Immunoprecipitation sequencing (ChIP-seq), RNA Sequencing (RNA-seq) and Assay for Transposase-Accessible Chromatin using sequencing (ATAC-seq) assays on wild-type strains and various engineered gene knockout or overexpression strains ([125], [126], [127]). Although a combination of direct TR binding interactions, differential gene expression, and chromatin accessibility profiles provides valuable insights into the white-opaque switching mechanics, the logic of the GRNs formed by the white-opaque TRs, and how they function to control phenotypic switching, remains largely unkown.

Molecular biology has been successful at identifying the functional components of cells. Genomes of many organisms have been sequenced. A majority of genes, mRNA transcripts, proteins, and metabolites, have been characterized and catalogued in dedicated databases. However, these results are insufficient to gain a system-level insight since they offer no convincing concepts for how the components interact simultaneously and dynamically to generate systematic properties of a cell ([128]). These limitations give rise to the field of systems biology. GRN reverse engineering has been a major challenge in the field, since the expression of genes in a cell is controlled by its GRN. Over the last two

decade, many state-of-the-art models have been developed to infer GRN topology using "omics" data and other prior knowledge ([129]). Even with these various models, the GRN inference still remains an underdetermined problem since the number of variables (i.e. uncertainty) in the GRN exceeds the number of implemented measurements ([130]). Furthermore, since current technologies enable determination of only specific components of a real-life GRN, it can be challenging to experimentally verify GRN topology inferred by models. Specifically, the physical interactions between TRs and their downstream genes can be relatively easily determined by ChIP-seq (Fig. 1.3), but the logic of these binding remains non-determined. To deal with the underdetermination problem and the lack of real-life network benchmarks, a majority of the existing models turn to in-silico (i.e. computer generated) datasets. Although the *in-silico* network benchmark generators, such as GNW ([114]), SynTReN ([115]), and DREAM ([131]), provide GRN structures on different connection distributions and corresponding expression datasets, they don't necessarily capture or represent the accurate behavior of an underlying real-life GRN. These barriers call for a more refined model, which could infer both the *in-silico* and real-life GRNs and make experimentally testable predictions.

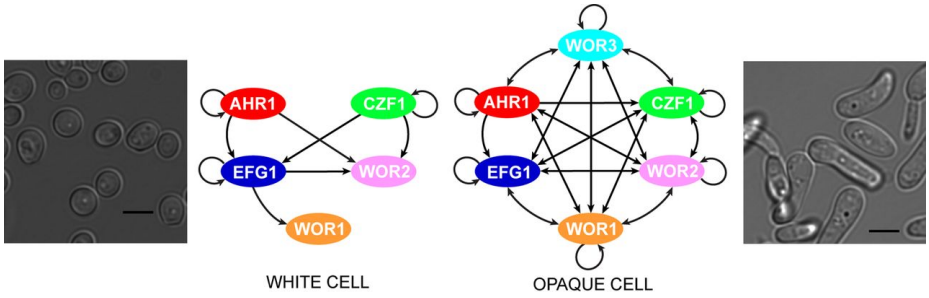


Figure 1.3: Physical binding map of the white and opaque core switch circuits, and the resulting cell types. Scale bar indicates 5 μm. Figure modified from [132]

.

1.5 Background of Problem

Although transcriptional profiles are informative and have been widely used, they do not directly reflect the regulatory status of a gene (i.e. whether a gene is activated or repressed) ([95]), since some mRNAs are highly stable and can accumulate in cells, while others are actively degraded. The fundamental criterion of how actively a gene is being transcribed should be the transcription rate ([96]). Specifically, Top-down high-throughput sequencing methods, such as ChIP-seq and RNA-seq, cannot directly measure

the regulatory interactions between TRs and DNA. In principle, RNA-seq only shows the relative abundance of transcripts with no information on physical interactions, while ChIP-seq captures the physical binding but does not reveal the regulatory effects. Alternatively, bottom-up synthetic techniques can be applied to genetic circuit engineering. The logic of the circuit can be designed by assembling well-characterized parts such as TR genes and their corresponding promoters. Nevertheless, the synthetic methods have their own weakness. For instance, the synthetic circuits are designed by human and therefore cannot reveal the unknown complexity of existing GRNs in various organisms.

1.6 Statement of Problem

The objectives of my study are:

(1) Develop a scalable GRN reverse engineering model that incorporates kinetic parameters and works on both in-silico and real-life datasets.

(2) Apply the model to the white-opaque switch core circuit in Candida albicans and make testable predictions on its behavior.

(3) Test the predictions experimentally with genome-editing tools and RNA sequencing and further refine the model.

1.7 Research Design

We first developed a scalable GRN inference model using a set of ordinary differential equations to simulate the processes involved in gene expression and an evolutionary algorithm to search for the GRN that best produces the target transcriptional profiles. Second, we evaluated its performance on existing *in-silico* and real-life datasets. Next, to decipher the GRN structure of the white-opaque switch circuit in *Candida albicans*, we performed RNA-seq and ChIP-seq on 40 different *Candida albicans* homozygous knockout strains, which were engineered by an efficient CRISPR-mediated genome-editing tool ([133]). In addition, we used csRNA-seq ([106]) to measure genome-wide transcription rates and mRNA degradation rates in *Candida albicans* wild types. Using the model and the data generated above, we obtained an inferred structure for the white-opaque circuit. with the inferred circuit topology, we used the model to introduce double knockouts and/or targeted disruptions to the important TR-DNA regulatory interactions and predicted how the steady-state transcriptional profiles were going to change. These predictions were tested later on by experimentally making the same disruptions and measuring the consequential

transcriptional profiles. The model predictions facilitated candidate selection for experimentation and the experimental results in turn provided validation or further improvement for the model.

Bibliography

[1] Blagoj Ristevski. A survey of models for inference of gene regulatory networks. *Nonlinear Analysis: Modelling and Control*, 18(4):444–465, 2013. ISBN: 1392-5113 Publisher: Vilnius University.

[2] Thuy Tien Bui and Kumar Selvarajoo. Attractor concepts to evaluate the transcriptome-wide dynamics guiding anaerobic to aerobic state transition in escherichia coli. *Scientific reports*, 10(1):1–14, 2020. ISBN: 2045-2322 Publisher: Nature Publishing Group.

[3] Pau Rué and Alfonso Martinez Arias. Cell dynamics and gene expression control in tissue homeostasis and development. *Molecular systems biology*, 11(2):792, 2015. ISBN: 1744-4292.

[4] Eric H. Davidson and Douglas H. Erwin. Gene regulatory networks and the evolution of animal body plans. *Science*, 311(5762):796–800, 2006. ISBN: 0036-8075 Publisher: American Association for the Advancement of Science.

[5] Tariq Enver, Martin Pera, Carsten Peterson, and Peter W. Andrews. Stem cell states, fates, and the rules of attraction. *Cell stem cell*, 4(5):387–397, 2009. ISBN: 1934-5909 Publisher: Elsevier.

[6] Stuart Kauffman. A proposal for using the ensemble approach to understand genetic regulatory networks. *Journal of theoretical biology*, 230(4):581–590, 2004. ISBN: 0022-5193 Publisher: Elsevier.

[7] Piyush B. Madhamshettiwar, Stefan R. Maetschke, Melissa J. Davis, Antonio Reverter, and Mark A. Ragan. Gene regulatory network inference: evaluation and application to ovarian cancer allows the prioritization of drug targets. *Genome Medicine*, 4(5):41, 2012-05-01.

[8] Daniel Moore, Ricardo de Matos Simoes, Matthias Dehmer, and Frank Emmert-Streib. Prostate cancer gene regulatory network inferred from RNA-seq data. *Current genomics*, 20(1):38–48, 2019. ISBN: 1389-2029 Publisher: Bentham Science Publishers.

[9] Johann Sebastian Hawe, Fabian J. Theis, and Matthias Heinig. Inferring interaction networks from multi-comics data-a review. *Frontiers in genetics*, 10:535, 2019. ISBN: 1664-8021 Publisher: Frontiers.

[10] Joseph L. Natale, David Hofmann, Damián G. Hernández, and Ilya Nemenman. Reverse-engineering biological networks from large data sets. *arXiv preprint arXiv:1705.06370*, 2017.

[11] Lars Kaderali and Nicole Radde. Inferring gene regulatory networks from expression data. In *Computational Intelligence in Bioinformatics*, pages 33–74. Springer, 2008.

[12] Alireza Fotuhi Siahpirani, Deborah Chasman, and Sushmita Roy. Integrative approaches for inference of genome-scale gene regulatory networks. In *Gene Regulatory Networks*, pages 161–194. Springer, 2019.

[13] Jimmy Omony. Biological network inference: A review of methods and assessment of tools and techniques. *Annual Research & Review in Biology*, pages 577–601, 2014. ISBN: 2347-565X.

[14] Wenting Liu and Jagath C. Rajapakse. Fusing gene expressions and transitive protein-protein interactions for inference of gene regulatory networks. *BMC systems biology*, 13(2):37, 2019. ISBN: 1752-0509 Publisher: Springer.

[15] Michael Hecker, Sandro Lambeck, Susanne Toepfer, Eugene Van Someren, and Reinhard Guthke. Gene regulatory network inference: data integration in dynamic models—a review. *Biosystems*, 96(1):86–103, 2009. ISBN: 0303-2647 Publisher: Elsevier.

[16] Guy Karlebach and Ron Shamir. Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology*, 9(10):770–780, 2008. ISBN: 1471-0080 Publisher: Nature Publishing Group.

[17] Blagoj Ristevski. Overview of computational approaches for inference of microrna-mediated and gene regulatory networks. In *Advances in Computers*, volume 97, pages 111–145. Elsevier, 2015.

[18] Wei-Po Lee and Wen-Shyong Tzou. Computational methods for discovering gene networks from expression data. *Briefings in bioinformatics*, 10(4):408–423, 2009. ISBN: 1477-4054 Publisher: Oxford University Press.

[19] Daniele Mercatelli, Laura Scalambra, Luca Triboli, Forest Ray, and Federico M. Giorgi. Gene regulatory network inference resources: A practical overview. *Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms*, 1863(6):194430, 2020. ISBN: 1874-9399 Publisher: Elsevier.

[20] Timothy S. Gardner and Jeremiah J. Faith. Reverse-engineering transcription control networks. *Physics of life reviews*, 2(1):65–88, 2005. ISBN: 1571-0645 Publisher: Elsevier.

[21] Michael Banf and Seung Y. Rhee. Computational inference of gene regulatory networks: approaches, limitations and opportunities. *Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms*, 1860(1):41–52, 2017. ISBN: 1874-9399 Publisher: Elsevier.

[22] Fernando M. Delgado and Francisco Gómez-Vela. Computational methods for gene regulatory networks reconstruction and analysis: A review. *Artificial intelligence in medicine*, 95:133–145, 2019. ISBN: 0933-3657 Publisher: Elsevier.

[23] Nooshin Omranian and Zoran Nikoloski. Computational approaches to study gene regulatory networks. In *Plant Gene Regulatory Networks*, pages 283–295. Springer, 2017.

[24] Nathalie Villa-Vialaneix, Matthieu Vignes, Nathalie Viguerie, and Magali San Cristobal. Inferring networks from multiple samples with consensus lasso. *Quality technology & quantitative management*, 11(1):39–60, 2014.

[25] Lide Han and Jun Zhu. Using matrix of thresholding partial correlation coefficients to infer regulatory network. *Biosystems*, 91(1):158–165, 2008.

[26] Adam A Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. In *BMC bioinformatics*, volume 7, pages 1–15. BioMed Central, 2006.

[27] Jamshid Pirgazi and Ali Reza Khanteymoori. A robust gene regulatory network inference method base on kalman filter and linear regression. *PLoS ONE*, 13, 2018.

[28] Scott Kirkpatrick, C. Daniel Gelatt, and Mario P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983. ISBN: 0036-8075 Publisher: American association for the advancement of science.

[29] Michael Maza and Deniz Yuret. Dynamic hill climbing. *AI Expert*, 9, 1994-01-01.

[30] William Chad Young, Adrian E. Raftery, and Ka Yee Yeung. Fast bayesian inference for gene regulatory networks using ScanBMA. *BMC systems biology*, 8(1):47, 2014. ISBN: 1752-0509 Publisher: Springer.

[31] Manuel Sanchez-Castillo, David Blanco, Isabel M. Tienda-Luna, M. C. Carrion, and Yufei Huang. A bayesian framework for the inference of gene regulatory networks from time and pseudo-time series data. *Bioinformatics*, 34(6):964–970, 2018. ISBN: 1367-4803 Publisher: Oxford University Press.

[32] Yue Fan, Xiao Wang, and Qinke Peng. Inference of gene regulatory networks using bayesian nonparametric regression and topology information. *Computational and mathematical methods in medicine*, 2017, 2017. ISBN: 1748-670X Publisher: Hindawi.

[33] Hirotaka Matsumoto and Hisanori Kiryu. SCOUP: a probabilistic model based on the ornstein–uhlenbeck process to analyze single-cell expression data during differentiation. *BMC bioinformatics*, 17(1):232, 2016. ISBN: 1471-2105 Publisher: Springer.

[34] David Maxwell Chickering. Learning bayesian networks is np-complete. *Learning from data: Artificial intelligence and statistics V*, pages 121–130, 1996.

[35] Frank Dondelinger, Dirk Husmeier, and Sophie Lèbre. Dynamic bayesian networks in molecular plant science: inferring gene regulatory networks from multiple gene expression time series. *Euphytica*, 183:361–377, 2012.

[36] Ildikó Flesch and Peter JF Lucas. Markov equivalence in bayesian networks. *Advances in probabilistic graphical models*, pages 3–38, 2007.

[37] Rui-Sheng Wang, Assieh Saadatpour, and Reka Albert. Boolean modeling in systems biology: an overview of methodology and applications. *Physical biology*, 9(5):055001, 2012. ISBN: 1478-3975 Publisher: IOP Publishing.

[38] Jan Krumsiek, Carsten Marr, Timm Schroeder, and Fabian J. Theis. Hierarchical differentiation of myeloid progenitors is encoded in the transcription factor network. *PloS one*, 6(8):e22649, 2011. ISBN: 1932-6203 Publisher: Public Library of Science.

[39] Stefan Bornholdt. Boolean network models of cellular regulation: prospects and limitations. *Journal of the Royal Society Interface*, 5:S85–S94, 2008. ISBN: 1742-5689 Publisher: The Royal Society London.

[40] Christoph Müssel, Martin Hopfensitz, and Hans A Kestler. Boolnet—an r package for generation, reconstruction and analysis of boolean networks. *Bioinformatics*, 26(10):1378–1380, 2010.

[41] Jie Zheng, David Zhang, Pawel F Przytycki, Rafal Zielinski, Jacek Capala, and Teresa M Przytycka. Simboolnet—a cytoscape plugin for dynamic simulation of signaling networks. *Bioinformatics*, 26(1):141–142, 2010.

[42] Clare E Giacomantonio and Geoffrey J Goodhill. A boolean model of the gene regulatory network underlying mammalian cortical area development. *PLoS computational biology*, 6(9):e1000936, 2010.

[43] Julio Saez-Rodriguez, Luca Simeoni, Jonathan A Lindquist, Rebecca Hemenway, Ursula Bommhardt, Boerge Arndt, Utz-Uwe Haus, Robert Weismantel, Ernst D Gilles, Steffen Klamt, et al. A logical model provides insights into t cell receptor signaling. *PLoS computational biology*, 3(8):e163, 2007.

[44] Rebekka Schlatter, Kathrin Schmich, Ima Avalos Vizcarra, Peter Scheurich, Thomas Sauter, Christoph Borner, Michael Ederer, Irmgard Merfort, and Oliver Sawodny. On/off and beyond-a boolean model of apoptosis. *PLoS computational biology*, 5(12):e1000595, 2009.

[45] Stuart A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3):437–467, 1969. ISBN: 0022-5193 Publisher: Academic Press.

[46] C. H. (Conrad Hal) Waddington. *The strategy of the genes : a discussion of some aspects of theoretical biology*. London : Routledge, 1957. Type: Book; Book/Illustrated.

[47] Akiko Kashiwagi, Itaru Urabe, Kunihiko Kaneko, and Tetsuya Yomo. Adaptive response of a gene network to environmental changes by fitness-induced attractor selection. *PloS one*, 1(1):e49, 2006. ISBN: 1932-6203 Publisher: Public Library of Science.

[48] Sui Huang, Gabriel Eichler, Yaneer Bar-Yam, and Donald E. Ingber. Cell fates as high-dimensional attractor states of a complex gene regulatory network. *Physical review letters*, 94(12):128701, 2005. Publisher: APS.

[49] Ahmadreza Ghaffarizadeh, Gregory J. Podgorski, and Nicholas S. Flann. Applying attractor dynamics to infer gene regulatory interactions involved in cellular differentiation. *Biosystems*, 155:29–41, 2017. ISBN: 0303-2647 Publisher: Elsevier.

[50] Stalin Munoz, Miguel Carrillo, Eugenio Azpeitia, and David A. Rosenblueth. Griffin: A tool for symbolic inference of synchronous boolean molecular networks. *Frontiers in genetics*, 9:39, 2018. ISBN: 1664-8021 Publisher: Frontiers.

[51] Ana Rodriguez, Isaac Crespo, Ganna Androsova, and Antonio del Sol. Discrete logic modelling optimization to contextualize prior knowledge networks using PRUNET. *PloS one*, 10(6):e0127216, 2015. ISBN: 1932-6203 Publisher: Public Library of Science San Francisco, CA USA.

[52] Arnaud Poret and Carito Guziolowski. Therapeutic target discovery using boolean network attractors: improvements of kali. *Royal Society open science*, 5(2):171852, 2018. ISBN: 2054-5703 Publisher: The Royal Society Publishing.

[53] Enrico Borriello, Sara I. Walker, and Manfred D. Laubichler. Cell phenotypes as macrostates of the GRN dynamics. *Journal of Experimental Zoology Part B: Molecular and Developmental Evolution*, 334(4):213–224, 2020. ISBN: 1552-5007 Publisher: Wiley Online Library.

[54] Daniel Marbach, Robert J. Prill, Thomas Schaffter, Claudio Mattiussi, Dario Floreano, and Gustavo Stolovitzky. Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of the national academy of sciences*, 107(14):6286–6291, 2010. ISBN: 0027-8424 Publisher: National Acad Sciences.

[55] Riet De Smet and Kathleen Marchal. Advantages and limitations of current network inference methods. *Nature Reviews Microbiology*, 8(10):717–729, 2010. ISBN: 1740-1534 Publisher: Nature Publishing Group.

[56] Atefeh Taherian Fard and Mark A. Ragan. Quantitative modelling of the waddington epigenetic landscape. In *Computational Stem Cell Biology*, pages 157–171. Springer, 2019.

[57] Gerhard Mayer. Modelling techniques for biomolecular networks. *arXiv preprint arXiv:2003.00327*, 2020.

[58] Timothy S. Gardner, Diego Di Bernardo, David Lorenz, and James J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301(5629):102–105, 2003. ISBN: 0036-8075 Publisher: American Association for the Advancement of Science.

[59] Siyuan Wu, Tiangang Cui, Xinan Zhang, and Tianhai Tian. A non-linear reverse-engineering method for inferring genetic regulatory networks. *PeerJ*, 8:e9065, 2020. ISBN: 2167-8359 Publisher: PeerJ Inc.

[60] Andrea Ocone, Laleh Haghverdi, Nikola S. Mueller, and Fabian J. Theis. Reconstructing gene regulatory dynamics from high-dimensional single-cell snapshot data. *Bioinformatics*, 31(12):i89–i96, 2015. ISBN: 1460-2059 Publisher: Oxford University Press.

[61] Richard Bonneau, David J. Reiss, Paul Shannon, Marc Facciotti, Leroy Hood, Nitin S. Baliga, and Vesteinn Thorsson. The inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome biology*, 7(5):R36, 2006. ISBN: 1474-760X Publisher: Springer.

[62] Baoshan Ma, Mingkun Fang, and Xiangtian Jiao. Inference of gene regulatory networks based on nonlinear ordinary differential equations. *Bioinformatics*, 2020.

[63] Chris J. Oates, Frank Dondelinger, Nora Bayani, James Korkola, Joe W. Gray, and Sach Mukherjee. Causal network inference using biochemical kinetics. *Bioinformatics*, 30(17):i468–i474, 2014. ISBN: 1460-2059 Publisher: Oxford University Press.

[64] Pierre Geurts. dynGENIE3: dynamical GENIE3 for the inference of gene networks from time series expression data. *Scientific reports*, 8(1):1–12, 2018. ISBN: 2045-2322 Publisher: Nature Publishing Group.

[65] Shuhei Kimura, Kaori Ide, Aiko Kashihara, Makoto Kano, Mariko Hatakeyama, Ryoji Masui, Noriko Nakagawa, Shigeyuki Yokoyama, Seiki Kuramitsu, and Akihiko Konagaya. Inference of s-system models of genetic networks using a cooperative coevolutionary algorithm. *Bioinformatics*, 21(7):1154–1163, 2005. ISBN: 1460-2059 Publisher: Oxford University Press.

[66] Josh Bevivino. The path from the simple pendulum to chaos. *Dynamics at the Horsetooth*, 1(1):1–24, 2009.

[67] Victor Olariu and Carsten Peterson. Kinetic models of hematopoietic differentiation. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 11(1):e1424, 2019. ISBN: 1939-5094 Publisher: Wiley Online Library.

[68] Leon Glass and Roderick Edwards. Hybrid models of genetic networks: Mathematical challenges and biological relevance. *Journal of theoretical biology*, 458:111–118, 2018. ISBN: 0022-5193 Publisher: Elsevier.

[69] Dominik M. Wittmann, Jan Krumsiek, Julio Saez-Rodriguez, Douglas A. Lauffenburger, Steffen Klamt, and Fabian J. Theis. Transforming boolean models to continuous models: methodology and application to t-cell receptor signaling. *BMC systems biology*, 3(1):98, 2009. ISBN: 1752-0509 Publisher: Springer.

[70] Nicolas Le Novère. Quantitative and logic modelling of molecular and gene networks. *Nature Reviews Genetics*, 16(3):146–158, 2015. ISBN: 1471-0064 Publisher: Nature Publishing Group.

[71] Jan Krumsiek, Sebastian Pölsterl, Dominik M. Wittmann, and Fabian J. Theis. Odefy-from discrete to continuous models. *BMC bioinformatics*, 11(1):233, 2010. ISBN: 1471-2105 Publisher: Springer.

[72] Daniel Aguilar-Hidalgo, María C. Lemos, and Antonio Córdoba. Evolutionary dynamics in gene networks and inference algorithms. *Computation*, 3(1):99–113, 2015. Publisher: Multidisciplinary Digital Publishing Institute.

[73] Joseph Xu Zhou, Lutz Brusch, and Sui Huang. Predicting pancreas cell fate decisions and reprogramming with a hierarchical multi-attractor model. *PloS one*, 6(3):e14752, 2011. ISBN: 1932-6203 Publisher: Public Library of Science.

[74] Jose Davila-Velderrain, Carlos Villarreal, and Elena R. Alvarez-Buylla. Reshaping the epigenetic landscape during early flower development: induction of attractor transitions by relative differences in gene decay rates. *BMC systems biology*, 9(1):1–14, 2015. ISBN: 1752-0509 Publisher: BioMed Central.

[75] Jing Guo, Feng Lin, Xiaomeng Zhang, Vivek Tanavde, and Jie Zheng. NetLand: quantitative modeling and visualization of waddington's epigenetic landscape using

probabilistic potential. *Bioinformatics*, 33(10):1583–1585, 2017. ISBN: 1367-4803 Publisher: Oxford University Press.

[76] Suoqin Jin, Fang-Xiang Wu, and Xiufen Zou. Domain control of nonlinear networked systems and applications to complex disease networks. *Discrete & Continuous Dynamical Systems-B*, 22(6):2169, 2017. Publisher: American Institute of Mathematical Sciences.

[77] Le-Zhi Wang, Ri-Qi Su, Zi-Gang Huang, Xiao Wang, Wen-Xu Wang, Celso Grebogi, and Ying-Cheng Lai. A geometrical approach to control and controllability of nonlinear dynamical networks. *Nature communications*, 7(1):1–11, 2016. ISBN: 2041-1723 Publisher: Nature Publishing Group.

[78] Atsushi Mochizuki, Bernold Fiedler, Gen Kurosawa, and Daisuke Saito. Dynamics and control at feedback vertex sets. II: A faithful monitor to determine the diversity of molecular activities in regulatory networks. *Journal of theoretical biology*, 335:130–146, 2013. ISBN: 0022-5193 Publisher: Elsevier.

[79] Jordan C. Rozum and Réka Albert. Identifying (un) controllable dynamical behavior in complex networks. *PLoS computational biology*, 14(12):e1006630, 2018. ISBN: 1553-734X Publisher: Public Library of Science San Francisco, CA USA.

[80] Jorge Gomez Tejeda Zañudo, Gang Yang, and Réka Albert. Structure-based control of complex networks with nonlinear dynamics. *Proceedings of the National Academy of Sciences*, 114(28):7234–7239, 2017. ISBN: 0027-8424 Publisher: National Acad Sciences.

[81] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. Controllability of complex networks. *nature*, 473(7346):167–173, 2011. ISBN: 1476-4687 Publisher: Nature Publishing Group.

[82] Atsushi Mochizuki. Theoretical approaches for the dynamics of complex biological systems from information of networks. *Proceedings of the Japan Academy, Series B*, 92(8):255–264, 2016. ISBN: 0386-2208 Publisher: The Japan Academy.

[83] James Henderson and George Michailidis. Network reconstruction using nonparametric additive ode models. *PloS one*, 9(4):e94003, 2014. ISBN: 1932-6203 Publisher: Public Library of Science.

[84] Leander Dony, Fei He, and Michael PH Stumpf. Parametric and non-parametric gradient matching for network inference: a comparison. *BMC bioinformatics*, 20(1):1–12, 2019. ISBN: 1471-2105 Publisher: BioMed Central.

[85] Benn Macdonald and Dirk Husmeier. Gradient matching methods for computational inference in mechanistic models for systems biology: a review and comparative analysis. *Frontiers in bioengineering and biotechnology*, 3:180, 2015. ISBN: 2296-4185 Publisher: Frontiers.

[86] Christopher A. Penfold, Iulia Gherman, Anastasiya Sybirna, and David L. Wild. Inferring gene regulatory networks from multiple datasets. In *Gene Regulatory Networks*, pages 251–282. Springer, 2019.

[87] Johannes Jaeger and Anton Crombach. Life's attractors : understanding developmental systems through reverse engineering and in silico evolution. *Advances in experimental medicine and biology*, 751:93–119, 2012. Place: United States.

[88] Tarmo Äijö and Harri Lähdesmäki. Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics. *Bioinformatics*, 25(22):2937–2944, 2009-08-25.

[89] Marcel H. Schulz, William E. Devanny, Anthony Gitter, Shan Zhong, Jason Ernst, and Ziv Bar-Joseph. DREM 2.0: Improved reconstruction of dynamic regulatory networks from time-series expression data. *BMC systems biology*, 6(1):104, 2012. ISBN: 1752-0509 Publisher: Springer.

[90] Sungjoon Park, Jung Min Kim, Wonho Shin, Sung Won Han, Minji Jeon, Hyun Jin Jang, Ik-Soon Jang, and Jaewoo Kang. BTNET: boosted tree based gene regulatory network inference algorithm using time-course measurement data. *BMC systems biology*, 12(2):69–77, 2018. ISBN: 1752-0509 Publisher: BioMed Central.

[91] Juan Camilo Castro, Ivan Valdés, Laura Natalia Gonzalez-García, Giovanna Danies, Silvia Cañas, Flavia Vischi Winck, Carlos Eduardo Ñústez, Silvia Restrepo, and Diego Mauricio Riaño-Pachón. Gene regulatory networks on transfer entropy (GRNTE): a novel approach to reconstruct gene regulatory interactions applied to a case study for the plant pathogen phytophthora infestans. *Theoretical Biology and Medical Modelling*, 16(1):7, 2019. ISBN: 1742-4682 Publisher: Springer.

[92] Wei Zhang, Wenchao Li, Jianming Zhang, and Ning Wang. Data integration of hybrid microarray and single cell expression data to enhance gene network inference. *Current Bioinformatics*, 14(3):255–268, 2019. ISBN: 1574-8936 Publisher: Bentham Science Publishers.

[93] Jamshid Pirgazi, Ali Reza Khanteymoori, and Maryam Jalilkhani. TIGRNCRN: Trustful inference of gene regulatory network using clustering and refining the network. *Journal of bioinformatics and computational biology*, 17(3):1950018, 2019. ISBN: 0219-7200 Publisher: World Scientific.

[94] Wenchao Li, Wei Zhang, and Jianming Zhang. A novel model integration network inference algorithm with clustering and hub genes finding. *Molecular Informatics*, 39(5):1900075, 2020. ISBN: 1868-1743 Publisher: Wiley Online Library.

[95] Simon J. Larsen, Richard Röttger, Harald H. H. W. Schmidt, and Jan Baumbach. E. coli gene regulatory networks are inconsistent with gene expression data. *Nucleic acids research*, 47(1):85–92, 2019. ISBN: 0305-1048 Publisher: Oxford University Press.

[96] Toshimichi Yamada and Nobuyoshi Akimitsu. Contributions of regulated transcription and mRNA decay to the dynamics of gene expression. *Wiley Interdisciplinary Reviews: RNA*, 10(1):e1508, 2019. ISBN: 1757-7004 Publisher: Wiley Online Library.

[97] Katsuyuki Yugi, Satoshi Ohno, James R. Krycer, David E. James, and Shinya Kuroda. Rate-oriented trans-omics: integration of multiple omic data on the basis of reaction kinetics. *Current Opinion in Systems Biology*, 15:109–120, 2019. ISBN: 2452-3100 Publisher: Elsevier.

[98] Ineke Brouwer and Tineke L. Lenstra. Visualizing transcription: key to understanding gene expression dynamics. *Current opinion in chemical biology*, 51:122–129, 2019. ISBN: 1367-5931 Publisher: Elsevier.

[99] Christian Miller, Björn Schwalb, Kerstin Maier, Daniel Schulz, Sebastian Dümcke, Benedikt Zacher, Andreas Mayer, Jasmin Sydow, Lisa Marcinowski, and Lars Dölken. Dynamic transcriptome analysis measures rates of mRNA synthesis and decay in yeast. *Molecular systems biology*, 7(1):458, 2011. ISBN: 1744-4292 Publisher: John Wiley & Sons, Ltd Chichester, UK.

[100] Hidenori Tani, Rena Mizutani, Kazi Abdus Salam, Keiko Tano, Kenichi Ijiri, Ai Wakamatsu, Takao Isogai, Yutaka Suzuki, and Nobuyoshi Akimitsu. Genome-wide determination of RNA stability reveals hundreds of short-lived noncoding transcripts in mammals. *Genome research*, 22(5):947–956, 2012. ISBN: 1088-9051 Publisher: Cold Spring Harbor Lab.

[101] Michal Rabani, Raktima Raychowdhury, Marko Jovanovic, Michael Rooney, Deborah J. Stumpo, Andrea Pauli, Nir Hacohen, Alexander F. Schier, Perry J. Blackshear, and Nir Friedman. High-resolution sequencing and modeling identifies distinct dynamic RNA regulatory strategies. *Cell*, 159(7):1698–1710, 2014. ISBN: 0092-8674 Publisher: Elsevier.

[102] Björn Schwalb, Margaux Michel, Benedikt Zacher, Katja Frühauf, Carina Demel, Achim Tresch, Julien Gagneur, and Patrick Cramer. TT-seq maps the human transient transcriptome. *Science*, 352(6290):1225–1228, 2016. ISBN: 0036-8075 Publisher: American Association for the Advancement of Science.

[103] Christian Riml, Thomas Amort, Dietmar Rieder, Catherina Gasser, Alexandra Lusser, and Ronald Micura. Osmium-mediated transformation of 4-thiouridine to cytidine as key to study RNA dynamics by sequencing. *Angewandte Chemie International Edition*, 56(43):13479–13483, 2017. ISBN: 1433-7851 Publisher: Wiley Online Library.

[104] Veronika A. Herzog, Brian Reichholf, Tobias Neumann, Philipp Rescheneder, Pooja Bhat, Thomas R. Burkard, Wiebke Wlotzka, Arndt von Haeseler, Johannes Zuber, and Stefan L. Ameres. Thiol-linked alkylation of RNA to assess expression dynamics. *Nature methods*, 14(12):1198–1204, 2017. ISBN: 1548-7105 Publisher: Nature Publishing Group.

[105] Jeremy A. Schofield, Erin E. Duffy, Lea Kiefer, Meaghan C. Sullivan, and Matthew D. Simon. TimeLapse-seq: adding a temporal dimension to RNA sequencing through nucleoside recoding. *Nature methods*, 15(3):221, 2018. ISBN: 1548-7105 Publisher: Nature Publishing Group.

[106] Sascha H. Duttke, Max W. Chang, Sven Heinz, and Christopher Benner. Identification and dynamic quantification of regulatory elements using total RNA. *Genome research*, 29(11):1836–1846, 2019. ISBN: 1088-9051 Publisher: Cold Spring Harbor Lab.

[107] Stefan Klumpp and Terence Hwa. Growth-rate-dependent partitioning of RNA polymerases in bacteria. *Proceedings of the National Academy of Sciences*, 105(51):20245–20250, 2008. ISBN: 0027-8424 Publisher: National Acad Sciences.

[108] Ahammed Sherief Kizhakkethil Youseph, Madhu Chetty, and Gour Karmakar. Reverse engineering genetic networks using nonlinear saturation kinetics. *Biosystems*, 182:30–41, 2019. ISBN: 0303-2647 Publisher: Elsevier.

[109] Kevin A. McGoff, Xin Guo, Anastasia Deckard, Christina M. Kelliher, Adam R. Leman, Lauren J. Francey, John B. Hogenesch, Steven B. Haase, and John L. Harer. The local edge machine: inference of dynamic models of gene regulation. *Genome biology*, 17(1):214, 2016. ISBN: 1474-760X Publisher: Springer.

[110] Javier González, Ivan Vujačić, and Ernst Wit. Inferring latent gene regulatory network kinetics. *Statistical applications in genetics and molecular biology*, 12(1):109–127, 2013. ISBN: 1544-6115 Publisher: De Gruyter.

[111] Faizan Ehsan Elahi and Ammar Hasan. A method for estimating hill function-based dynamic models of gene regulatory networks. *Royal Society open science*, 5(2):171226, 2018. ISBN: 2054-5703 Publisher: The Royal Society Publishing.

[112] Alejandro F. Villaverde and Julio R. Banga. Reverse engineering and identification in systems biology: strategies, perspectives and challenges. *Journal of the Royal Society Interface*, 11(91):20130505, 2014. ISBN: 1742-5689 Publisher: The Royal Society.

[113] Maria I. Arnone and Eric H. Davidson. The hardwiring of development: organization and function of genomic regulatory systems. *Development*, 124(10):1851–1864, 1997. ISBN: 0950-1991 Publisher: The Company of Biologists Ltd.

[114] Thomas Schaffter, Daniel Marbach, and Dario Floreano. GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics*, 27(16):2263–2270, 2011. ISBN: 1460-2059 Publisher: Oxford University Press.

[115] Tim Van den Bulcke, Koenraad Van Leemput, Bart Naudts, Piet van Remortel, Hongwu Ma, Alain Verschoren, Bart De Moor, and Kathleen Marchal. SynTReN: a generator of synthetic gene expression data for design and analysis of structure

learning algorithms. *BMC bioinformatics*, 7(1):43, 2006. ISBN: 1471-2105 Publisher: Springer.

[116] Pedro Mendes, Wei Sha, and Keying Ye. Artificial gene networks for objective comparison of analysis algorithms. *Bioinformatics*, 19:ii122–ii129, 2003. ISBN: 1460-2059 Publisher: Oxford University Press.

[117] Diogo Camacho, PAOLA VERA LICONA, Pedro Mendes, and Reinhard Laubenbacher. Comparison of reverse-engineering methods using an in silico network. *Annals of the New York Academy of Sciences*, 1115(1):73–89, 2007. ISBN: 0077-8923 Publisher: Wiley Online Library.

[118] Barbara Di Camillo, Gianna Toffolo, and Claudio Cobelli. A gene network simulator to assess reverse engineering algorithms. *Annals of the New York Academy of Sciences*, 1158(1):125–142, 2009. ISBN: 0077-8923 Publisher: Wiley Online Library.

[119] Daniel Marbach, Thomas Schaffter, Claudio Mattiussi, and Dario Floreano. Generating realistic in silico gene networks for performance assessment of reverse engineering methods. *Journal of Computational Biology*, 16(2):229–239, 2009. Num Pages: 229-239.

[120] Daniel Marbach, James C. Costello, Robert Küffner, Nicole M. Vega, Robert J. Prill, Diogo M. Camacho, Kyle R. Allison, Manolis Kellis, James J. Collins, and Gustavo Stolovitzky. Wisdom of crowds for robust gene network inference. *Nature methods*, 9(8):796–804, 2012. ISBN: 1548-7105 Publisher: Nature Publishing Group.

[121] Norma V Solis, Yang-Nim Park, Marc Swidergall, Karla J Daniels, Scott G Filler, and David R Soll. Candida albicans white-opaque switching influences virulence but not mating during oropharyngeal candidiasis. *Infection and immunity*, 86(6):10–1128, 2018.

[122] B Slutsky, M Staebell, J Anderson, L Risen, M t Pfaller, and DR Soll. " white-opaque transition": a second high-frequency switching system in candida albicans. *Journal of bacteriology*, 169(1):189–197, 1987.

[123] Matthew B Lohse and Alexander D Johnson. White–opaque switching in candida albicans. *Current opinion in microbiology*, 12(6):650–654, 2009.

[124] Corey Frazer, Aaron D Hernday, and Richard J Bennett. Monitoring phenotypic switching in candida albicans and the use of next-gen fluorescence reporters. *Current protocols in microbiology*, 53(1):e76, 2019.

[125] Rebecca E Zordan, Mathew G Miller, David J Galgoczy, Brian B Tuch, and Alexander D Johnson. Interlocking transcriptional feedback loops control white-opaque switching in candida albicans. *PLoS biology*, 5(10):e256, 2007.

[126] Aaron D Hernday, Matthew B Lohse, Polly M Fordyce, Clarissa J Nobile, Joseph L DeRisi, and Alexander D Johnson. Structure of the transcriptional network controlling white-opaque switching in c andida albicans. *Molecular microbiology*, 90(1):22–35, 2013.

[127] Matthew B Lohse, Iuliana V Ene, Veronica B Craik, Aaron D Hernday, Eugenio Mancera, Joachim Morschhäuser, Richard J Bennett, and Alexander D Johnson. Systematic genetic screen for transcriptional regulators of the candida albicans white-opaque switch. *Genetics*, 203(4):1679–1692, 2016.

[128] Uwe Sauer, Matthias Heinemann, and Nicola Zamboni. Getting closer to the whole picture. *Science*, 316(5824):550–551, 2007.

[129] Ziv Bar-Joseph, Georg K Gerber, Tong Ihn Lee, Nicola J Rinaldi, Jane Y Yoo, François Robert, D Benjamin Gordon, Ernest Fraenkel, Tommi S Jaakkola, Richard A Young, et al. Computational discovery of gene modules and regulatory networks. *Nature biotechnology*, 21(11):1337–1342, 2003.

[130] Riet De Smet and Kathleen Marchal. Advantages and limitations of current network inference methods. *Nature Reviews Microbiology*, 8(10):717–729, 2010.

[131] Gustavo Stolovitzky, DON Monroe, and Andrea Califano. Dialogue on reverse-engineering assessment and methods: the dream of high-throughput pathway inference. *Annals of the New York Academy of Sciences*, 1115(1):1–22, 2007.

[132] Aaron D Hernday, Matthew B Lohse, Clarissa J Nobile, Liron Noiman, Clement N Laksana, and Alexander D Johnson. Ssn6 defines a new level of regulation of white-opaque switching in candida albicans and is required for the stochasticity of the switch. *MBio*, 7(1):e01565–15, 2016.

[133] Valmik K Vyas, M Inmaculada Barrasa, and Gerald R Fink. A candida albicans crispr system permits genetic engineering of essential genes and gene families. *Science advances*, 1(3):e1500248, 2015.

CHAPTER II

MODEL FORMULATION

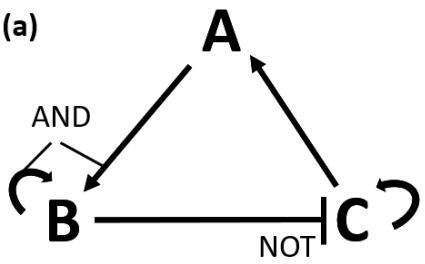
In this chapter, we built an ODE-based model to simulate the biological processes of transcription, translation, and TF regulation. In our model, the architecture of the GRN determines the ODE formulation and the ODEs derive the dynamics of gene expression. To reversely infer a GRN from transcriptional profiles as dynamical stable states, we wrote an evolutionary algorithm which can select a GRN from a population that can best reproduce the anticipated stable states via the ODE-based model.

2.1 GRN architecture depiction

Drawing on the conventions of early work ([1]), we depict the GRN architecture as a directed graph consisting of nodes representing genes and TFs, and edges representing interactions among these nodes (Fig. 2.1). For example, a simple GRN architecture is shown in Fig. 2a. In this graph, the nodes A, B, and C, represent the three genes in the GRN and the TFs they encode, respectively. Three types of interactions exist between these TFs and genes: activating, inhibitory, or neutral, and they are represented by pointed, blunt, or no arrows in the graph. In addition to the TF-gene interactions, TF-TF ones also exist in the GRN architecture, and they are represented by the combinatorial logic operators (AND, OR, or NOT), which indicate how the TFs work together to regulate their target gene. As shown in Fig. 2.1b and 2.1c, the TF-gene relations are denoted by an adjacency matrix named AM, which uses 1, -1, and 0 to indicate the pointed, blunt, and no arrows respectively in the directed graph. The TF-TF interactions are organized in another matrix denoted by LG, whose Boolean values are assigned to decide whether the activators, or repressors, of a gene work synergistically or independently. We use f_0, which is bounded

between 0 and 1, to represent the basal expression level of a gene when no TF acting on it. The overall GRN architecture, or A_{net}, can be expressed by $AM_{n\times n}$ $LG_{2\times n}$, and f_0, where n denotes the number of genes in the GRN. In this simplified GRN architecture, the activators or repressors of a gene have to work either synergistically as a complex or independently as monomers (Fig. 2.1c). This simplified network depiction preserves the two typical ways of how multiple activators or repressors can work and largely reduces complexity in network logic gates (2^{2N} potential values), while it leaves the GRN topology a relatively large space to change (3^{N^2} potential values).

(a)



(b)

Adjacency Matrix (AM)				
		Target Genes		
		A	B	C
TFs	A	0	1	0
	B	0	1	-1
	C	1	0	1

(c)

Protein Coordination Matrix (LG)			
	A	B	C
ActivatorNmer	0	1	0
RepressorNmer	0	0	0
(Basal) f_0	[0,1]	[0,1]	[0,1]

Figure 2.1: Depiction of a hypothetical GRN architecture. (a) Schematic of a simple GRN in which A and B cooperatively activate B, C activates A and itself, and B represses C in a manner that can override the self-activation of C. (b) The network topology table represents the direct activating, inhibiting, and null connections by 1, -1, and 0, respectively. (c) The protein coordination parameters are assigned to each gene in the genome and qualitatively describe the coordination between each gene's regulatory TFs. 'ActivatorNmer' decides whether the activators of a gene work independently (0) or cooperatively (1); 'RepressorNmer' decides whether the repressors of a gene work independently (0) or cooperatively (1). f_0 determines the basal expression level of a gene and whether its activators or repressors outcompete the other.

2.2 GRN dynamical system

A list of symbols and parameters used in the GRN dynamic system is given in Table 2.1.

Table 2.1: Parameter table for the GRN dynamic system

Symbol	Description	Unit
$[R]_i$	Number of mRNA transcripts for each gene	Dimensionless
$[P]_i$	Number of protein copies encoded by each gene	Dimensionless
$V_{i,max}$	Maximal rate of transcription for each promoter	Nucleotides/second
$V_{i,min}$	Minimal rate of transcription for each promoter	Nucleotides/second
$f_{0,i}$	Basal expression of a gene: a percentage of the $V_{i,max}$	Percentage
$V_{i,trl}$	Rate of translation for each proteins	Amino acids/second
$T_{i,j}$	Also known as K_A, the protein abundance producing half occupation	Dimensionless
k_i	Hill coefficient	Dimensionless
A_{net}	The architecture of a GRN, including the adjacency matrix and the logic gates matrix	Dimensionless
AM	The adjacency matrix of a GRN	Dimensionless
LG	The logic gates matrix of a GRN	Dimensionless
$D_{i,mRNA}$	Rate of degradation for each mRNA	1/second
$D_{i,protein}$	Rate of degradation for each protein	1/second
$I_{n\times m}$	The input matrix that contains m steady-state transcription profiles in the length of n genes	Dimensionless

We assume that the diffusion and TF binding processes are instantaneous: they happen much faster than transcription and translation and can be ignored. We assign a V_{max} and a V_{min} to each gene, which represent the potential highest and lowest production rates of mRNA transcripts, respectively. These two mRNA production rates are believed to be intrinsic properties of the target genes' promoters and independent to the regulatory statuses (i.e. promoter states) of the genes. In other words, the TFs cannot change the V_{max}

and V_{min} of the target gene. We believe that it is the $(V_{max} - V_{min})$ on which the TFs act to control the activity of a promoter (Eq. II.1). The process of TF regulation is represented by a function called the regulation function, which is denoted by $f_{A_{net}}$. For example, when the target gene is fully inhibited, the regulation function equals 0 and the production rate of the gene will be its leakage rate. With these settings, we formulate the difference equations of $[R]$ and $[P]$ as Eq. II.1 and II.2. Given the initial state, we could perform numerical solution to calculate the $[R]$ and $[P]$ at the next time point.

$$\begin{aligned} \frac{\Delta[R]_i}{\Delta t} &= V_{i,min} + (V_{i,max} - V_{i,min}) \cdot f_{A_{net}}([P]_{*,i}, \Theta) - D_{mRNA} \cdot [R]_i \\ \Theta &= \{f_0, T, k\} \\ A_{net} &= \{AM, LG\}, \end{aligned} \tag{II.1}$$

$$\frac{\Delta[P]_i}{\Delta t} = V_{trl} \cdot [R]_i - D_{protein} \cdot [P]_i, \tag{II.2}$$

where $[P]_{*,i}$ means the $[P]$ of all the TFs that regulate the i^{th} gene in a GRN, and $f_{A_{net}}$ is the regulation function which determines how TFs regulate a gene. Other symbols in Eq. II.1 and II.2 have already been defined in Table 2.1. The regulation function $f_{A_{net}}$ is a continuous function given in Eq. 7. Typically, Hill function ([2, 3, 4]) and sigmoid function ([5, 6]) have been applied in the regulation function. In this model we apply a modified Hill function (Eq. II.3 and II.4) to formulate the regulation function $f_{A_{net}}$.

$$S_{act,i}([P]_i) = \frac{[P]_i^{k_i}}{[P]_i^{k_i} + T_i^{k_i}}, \tag{II.3}$$

$$S_{rep,i}([P]_i) = 1 - S_{act,i}([P]_i), \tag{II.4}$$

where $[P]$ is the TF concentration; T, also known as K_A, is the protein abundance producing half occupation; k is the hill coefficient. Eq. II.3 is the Hill function for activators and Eq. II.4 is for repressors. When the activators (or repressors) are present, or $[P] - T > 0$, the increment in their $[P]$ will increase (or decrease) the transcription rate of the target gene. When a gene is regulated by multiple TFs, the interactions between TFs need to be considered. Activators and repressors can form polymers or simply work as monomers. Whether the activators and repressors work synergistically or independently for each gene is defined by the LG. If a gene has both activator and repressor simultaneously, the binding

competition between the two types of regulators have to be determined by an additional , f_0.

Different function formulas, as listed in Table 2.2, are used according to how the TFs work defined by LG. The combinatorial functions satisfy the following conditions:

(i) All four functions are bounded between 0 and 1 for positive TF values.

(ii) Synergistic activation requires more than none of the activators are absent.

(iii) Independent activators give a value close to one even if only one $S_{act,i}([P]_i)$ is close to one.

(iv) If a single repressor is absent, the synergistic repressor function is close to one.

(v) If even one repressor is high, then the independent repressor function is close to one.

Table 2.2: Combinatorial Hill functions for multiple TF regulation

Multiple TF regulation	Combinatorial Hill function
Independent activators	$C_{IA,i}([P]_i) = 1 - \prod_{i=1}^{numTF} (1 - S_{act,i}([P]_i))$
Independent repressors	$C_{IR,i}([P]_i) = \prod_{i=1}^{numTF} (1 - S_{act,i}([P]_i))$
Synergistic activators	$C_{SA,i}([P]_i) = \prod_{i=1}^{numTF} S_{act,i}([P]_i)$
Synergistic repressors	$C_{SR,i}([P]_i) = 1 - \prod_{i=1}^{numTF} S_{act,i}([P]_i)$

The effects of the combinatorial functions are shown in Fig. 2.2, where two activators, or repressors, regulate a gene synergistically as dimers, or independently as monomers. It can be seen that when the two TFs work synergistically as dimers (Fig. 2.2c and d), their $[P]$ have to be both high to significantly change the value of $f_{A_{net}}$. Contrarily, when they work as monomers (Fig. 2.2a and b), either TF can change the value of $f_{A_{net}}$ without the other.

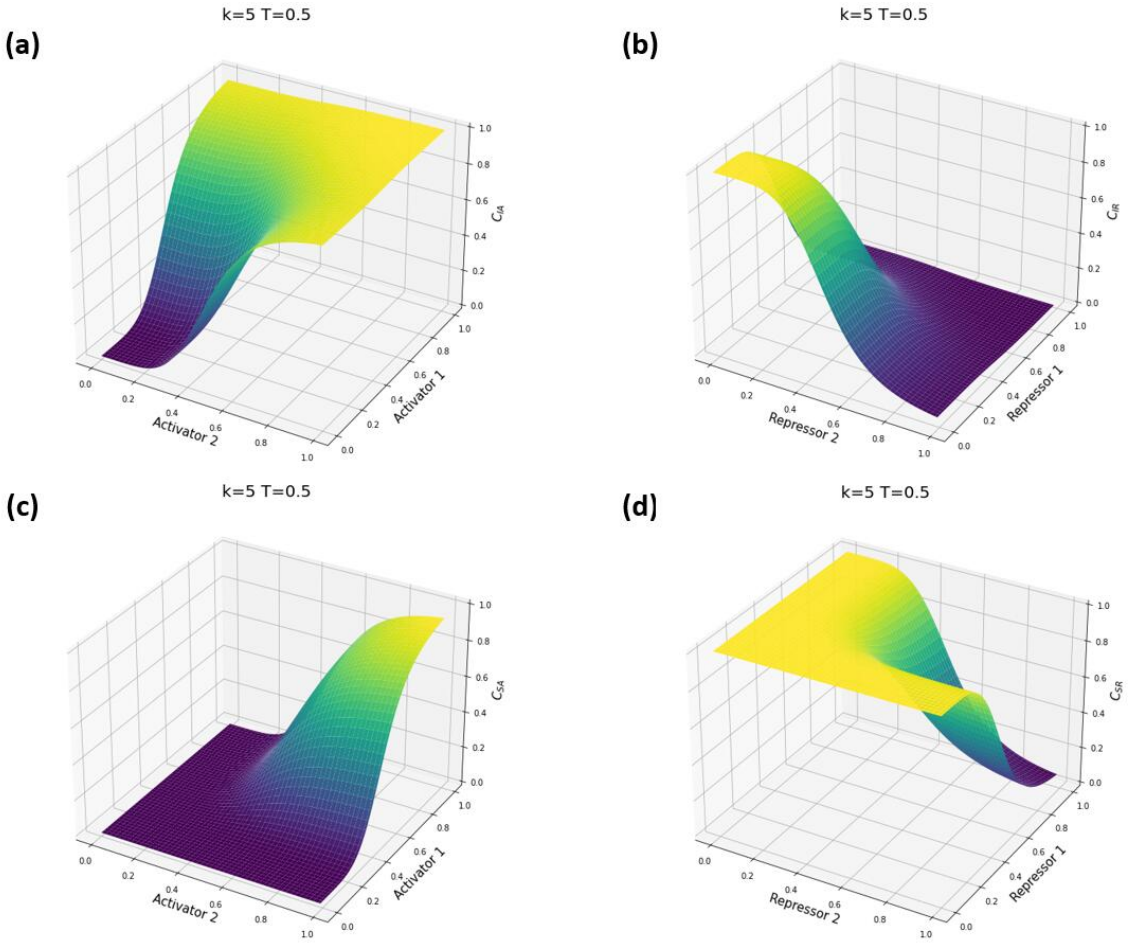


Figure 2.2: Demonstration of the effects of simulating multiple TFs regulation using the formulas in Table 2.2. The x-axis and y-axis are the protein concentration of the two activators or repressors, and the z-axis shows the outcome of the combinatorial functions. (a) Two activators work independently as monomer to activate a target gene. (b) Two repressors work independently as monomer to inhibit a target gene. (c) Two activators work synergistically as dimer. (d) Two repressors work synergistically as dimer.

With the A_{net} and the abundances of the TFs, we define the regulation function $f_{A_{net}}$ by Eq. II.5, which applies combinatorial Hill function formulas (Table 2.2) and f_0 to represent the gene regulations.

$$f_{A_{net}}([P]_{*,i}, \Theta) = f_0 + f_0 \cdot (C_A - 1) \cdot (1 - C_R) + (1 - f_0) \cdot C_A \cdot C_R, \tag{II.5}$$

where $[P]_{*,i}$ denotes the effective abundance of the TFs regulating the i^{th} gene. For notational convenience, we define $C_{A,i} = C_{SA,i}$ if activators are synergistic and $C_{A,i} = C_{IA,i}$ if they

are independent, and similarly for $C_{R,i}$. How the parameters affect the shape of $f_{A_{net}}$ is shown in Fig. 2.3.

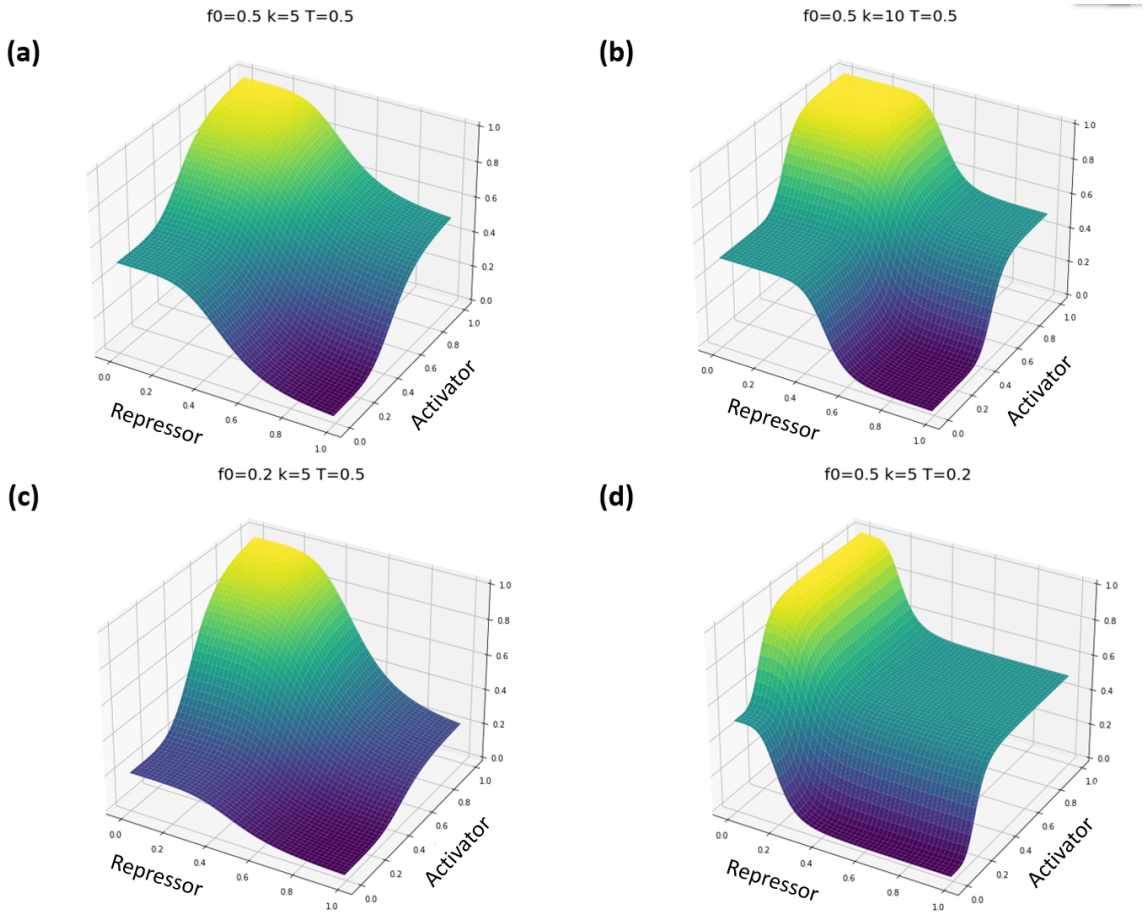


Figure 2.3: Demonstration of the regulation function $f_{A_{net}}$. The x-axis and y-axis are the $[P]$ of the repressors and activators, and the z-axis shows the outcome of the $f_{A_{net}}$. (a) $f_{A_{net}}$ has two fixed points: (1,0,0) and (0,1,1): the target gene is fully activated/inhibited when it has excess activators/repressors. (b) The Hill coefficient, k, determines the steepness of the regulation function. (c) The basal expression level, f_0, controls the position of the middle plane and can slide between 0 and 1. (d) The threshold T decides the TF abundance that will trigger the activation or repression.

2.3 Steadiness, stability, and attractor of a GRN system

Since we limit our simulation to the dynamics of transcription and translation processes, we describe the state of a cell, or equivalently a GRN system, as the abundance of mRNA transcripts and proteins: $[R]$ and $[P]$ (Fig. 2.4). Similar to a single pendulum (Fig. 2.4a), a GRN system could be at steady states (point Q), where the $[R]$ and $[P]$ do not change over time, or stable states (point P), where the system tends to approach and stabilize at (Fig.

2.4c and S1d). The steady-state definition is given by Eq. II.6 and II.7:

$$\frac{d[R]_{ss}}{dt} = V_{trc} - [R]_{ss} \cdot D_{mRNA} = 0, \tag{II.6}$$

$$\frac{d[P]_{ss}}{dt} = V_{trl} - [P]_{ss} \cdot D_{protein} = 0, \tag{II.7}$$

where $[R]_{ss}$ and $[P]_{ss}$ are mRNA and protein abundance at steady states; V_{trc} and V_{trl} are rates of transcription and translation; D_{mRNA} and $D_{protein}$ are degradation rates of mRNA and protein. In a kinetic system, steadiness does not necessarily imply stability, and vice versa. For instance, if a pendulum at steady state Q is slightly perturbed, it will fall and never reach Q again if friction exists. A system state that is both steady and stable, such as point P, is known as a fixed-point attractor. Supported by the previous studies ([7, 8, 9, 10]), we consider the input steady-state transcription profiles as fixed-point or limit cycle attractors in the GRN dynamic space. When perturbed by temperature, pH, or other environmental stimulus, a cellular state would be deviated from the stable state. If the stimuli disappears, the cellular state, constrained by the GRN dynamic system, would return to the original stable state, or fall into another stable state if the perturbation has driven it out of the basin of the first attractor. Below is an experimental evidence supporting the assumption that steady-state transcription profiles are attractors.

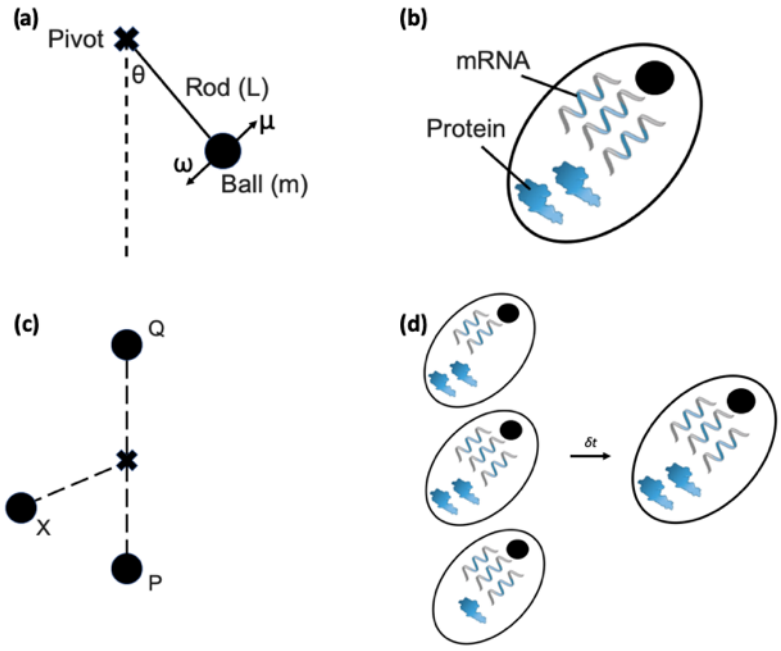


Figure 2.4: Side-by-side comparison of a single pendulum system and a cellular system. (a) Schematic of a simple pendulum model, which is a massive ball (m) connected to a pivot by a massless rod (L). (b) Schematic of a GRN system, which is described by its state variables: [R] and [P]. (c) State X is an arbitrary unsteady and unstable state of the pendulum system. State Q is steady but not stable. State P is both steady and stable, which makes it a fixed-point attractor. (d) Like the single pendulum, cells at steady and stable state (point P) can stay the same over time and resist mild perturbation.

In the experiment conducted by ([7]), the HL60 cells from a human leukemia cell line, were treated with two chemical compounds, all-trans retinoid acid and dimethylsulfoxide, at time 0 to induce differentiation towards neutrophil-like cells (Fig. 2.5). Selected gene expression profile snapshots along the two differentiation trajectories are shown as the heat maps. These heat maps show that after 168 hours, the two expression profiles, consisting of approximately 3000 genes, converged to a very similar pattern. The result indicates that the HL60 and the neutrophil-like cell type are like fixed-point attractors, towards which the intermediate and unstable cells would evolve. Under all-trans retinoid acid and dimethylsulfoxide, the cells were deviated out of the basin of the HL60 cell type attractor. They finally fell into the neutrophil-like cell type attractor through two different paths.

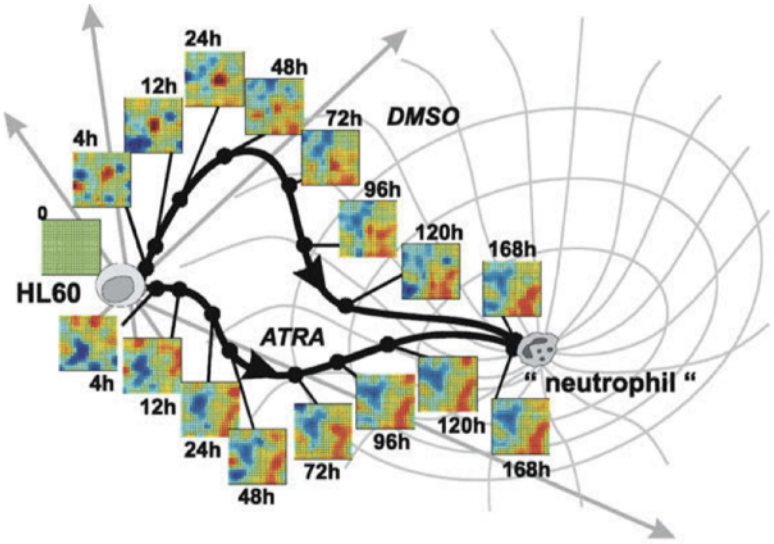


Figure 2.5: Experimental evidence that a cell type is a high-dimensional stable attractor in gene expression state space. The HL60 cells were treated with all-trans retinoid acid (ATRA) and dimethylsulfoxide (DMSO), respectively. They triggered cell differentiation towards the neutrophil-like cell type in two different trajectories. Selected gene expression profiles along the trajectories are shown in the heat maps. Modified from original paper ([7]).

2.4 GRN architecture inference: the evolutionary algorithm

With the deterministic GRN dynamic model constructed above, we propose to infer the A_{net} using experimentally-derived transcriptional profiles and mRNA production rates. Specifically, we consider transcriptional profiles of cells in the exponential growth phase under defined and mixed culture conditions. We therefore assume that the resulting transcriptional profiles represent a steady-state transcriptional output of the GRN. By incorporating experimentally determined transcription, translation, and degradation rates, we simulate the GRN dynamics and determine whether a given A_{net} can accurately reproduce the observed attractors. To search for the optimal A_{net} for a particular GRN, we utilize a modified evolutionary algorithm [11, 12, 13] to iteratively refine the A_{net} parameters until the predicted network attractors converge upon the experimentally measured ones. The main step-by-step processes of our iterative computational and experimental strategy are presented in Fig 2.6.

Due to the fact that the A_{net} and the values of some system parameters are often unknown in practice, we are going to make use of the measurable kinetic parameters (including V_{max}, V_{trl}, D_{mRNA}, and $D_{protein}$) and the steady-state transcription profiles to estimate the unknown parameters, which are V_{min}, T, f_0, and k. Since the transcription

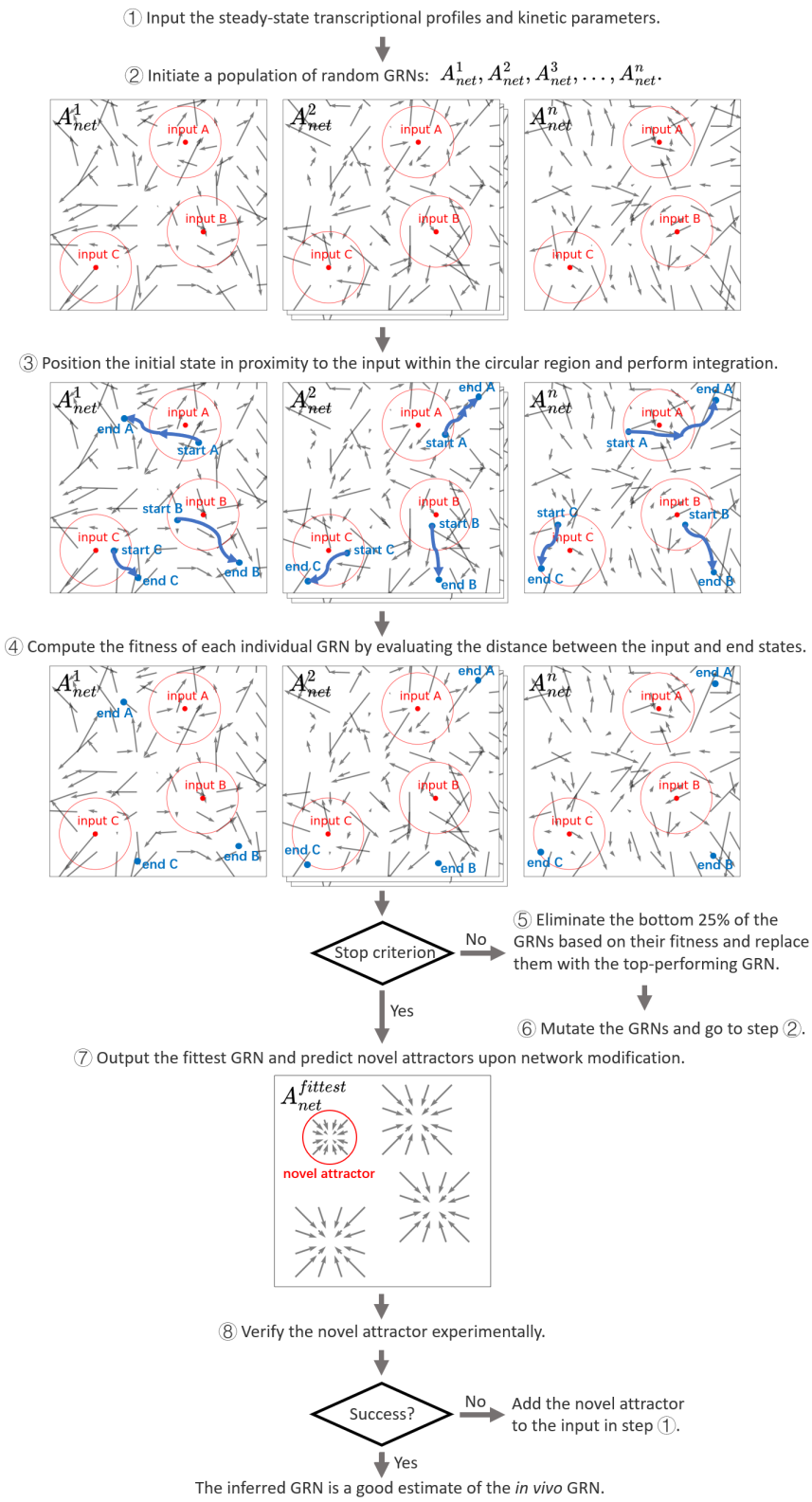


Figure 2.6: A flowchart illustrating the step-by-step processes of our iterative computational and experimental strategy to infer GRNs and predict novel attractors.

profiles are assumed to be steady states, we apply Eq. II.6 and II.7 to all the gene expression levels in the $I_{n\times m}$. First, the V_{min} of a gene is estimated by the minimal expression level of the gene across all the samples, as shown in Eq. II.8 below.

$$V_{i,min} - \min(I_{i,*}) \cdot D_{i,mRNA} = 0. \tag{II.8}$$

Therefore, given the $I_{n\times m}$ and D_{mRNA}, we can calculate the V_{min} using Eq. II.9.

$$V_{i,min} = \min(I_{i,*}) \cdot D_{i,mRNA}. \tag{II.9}$$

Second, without other prior knowledge, we have to assume that when the genes are under TF regulation, they have the same chance to be activated or inhibited. Hence, the T of the TFs is calculated by their average expression levels using Eq. II.2, which leads to Eq. II.10.

$$T_{*,i} = \frac{\frac{1}{2} \cdot (max(I_{i,*}) + min(I_{i,*})) \cdot V_{i,trl}}{D_{i,protein}}. \tag{II.10}$$

Third, we assume that the number of input transcription profiles is sufficient and the expression levels of fully activated, or inhibited genes were included in the observed data. When $[P] = [P]_{max}$, the outcome of Eq. II.11, denoted by q, should be close to 1. Therefore, we can calculate the Hill coefficient using Eq. II.12.

$$\frac{[P]^k}{[P]^k + T^k} = q. \tag{II.11}$$

$$k = \frac{\log \frac{1-q}{q}}{\log \frac{T_i}{[P]}}. \tag{II.12}$$

$[P]_{max}$ can be derived from Eq. II.6 and II.7 based on the steady-state hypothesis.

$$[P]_{max} = \frac{V_{trl} \cdot max(I_{i,*})}{D_{protein}}. \tag{II.13}$$

We have estimated the V_{min}, T, and k by $I_{n\times m}$ and other known parameters. The last unknown parameter, f_0, is inferred along with the A_{net}. Under the steady-state hypothesis, the derivative $\frac{\Delta[R]}{\Delta t}$ in Eq. II.1 equals 0. We can rewrite Eq. II.1 to calculate the f_0:

$$f_0 = \frac{\frac{D_{i,mRNA} \cdot I_{i,j} - V_{i,min}}{V_{i,max} - V_{i,min}} - C_A \cdot C_R}{C_A + C_R - 2 \cdot C_A \cdot C_R}. \tag{II.14}$$

In order to efficiently search through the extremely large GRN architecture space looking for the one that leads to the expected attractors (i.e. attractor matching), we propose a modified evolutionary algorithm as illustrated in Algorithm 1. First, we randomly create a population of A_{net}, each of which has the same initial fitness score. Second, we add a mild perturbation to each input transcription profile, or attractor, in the $I_{n\times m}$. We set them as initial states and run the deterministic GRN dynamic system by Runge-Kutta 4^{th} order ([14]) numerical solution to find the final steady states. See appendix A for detailed information about numerical solutions. Since we consider the transcription profiles in $I_{n\times m}$ as attractors, when the GRN system state is initiated nearby, it should be attracted to the attractors' positions. If the system state ends up being far away from the attractors, the current A_{net} cannot generate the attractors in the $I_{n\times m}$. We use a distance between the attractors and the final states as a metric:

$$AttractorDistance_j = \frac{1}{n} \cdot \sum_{i=1}^{n} \frac{|state_{i,j} - I_{i,j}|}{\max(I_{i,*})}, \tag{II.15}$$

where $state_{i,j}$ is the i^{th} gene expression level of the j^{th} final state obtained by running the GRN dynamic system, and $I_{i,j}$ is the i^{th} value of the j^{th} attractor in the $I_{n\times m}$. m is the total number of attractors and n is the total number of genes. We define the best A_{net} in the current population by the overall minimal "AttractorDistance". Third, we create the next generation population by mutating each A_{net} (by a certain Hamming distance, see appendix), and obtain the new minimal "AttractorDistance". If the new minimal "AttractorDistance" is less, we keep the mutated population. Otherwise, the mutated population will be abandoned and we will return to the former population. Next, we update the fitness of each A_{net} according to their "AttractorDistance". We sort the population by fitness in descending order and eliminate the last 20% individuals and we duplicate the first, or the fittest, individual to fill up the vacancy in population. By iteratively running the algorithm, we can obtain the fittest A_{net} in the last generation as the output.

Algorithm 1 Evolutionary Algorithm

Randomly initiate a population $\mathbf{A^0_{net}} : \{A^1_{net},\ A^2_{net},\ ...,\ A^N_{net}\}$;
Each individual has an initial $fitness : \frac{1}{N}$
$Calculate\ f_0\ for\ each\ A_{net}$
for $i\ in\ 1 : m$ **do**
  $initial\ system\ state \leftarrow I_{n,i}$
  $update\ f_0\ for\ each\ A_{net}\ and\ add\ penalty\ when\ f_0\ \notin [0, 1]$
  if A_{net} *has an independent* $self$ *activating edge* & *the initial system state in which the* $self$ *activating gene has been set to its maximal expression* $\notin I_{n,m}$ **then**
    $append\ the\ modified\ state\ described\ above\ to\ I_{n,m}$
  end if
  $initial\ system\ state \leftarrow initial\ system\ state\ + mild\ random\ perturbation$
  for $each\ A^j_{net}$ **do**
    $integrate\ GRN\ ODEs\ with\ A^j_{net} : \{AM_j,\ LG_j,\ f_{0j}\}$
  end for
  $record\ last\_states_{i,j}$
end for
$Calculate\ \min(AttractorDistance_{\mathbf{A^0_{net}}})$

for $generation\ from\ 1\ to\ x$ **do**
  for $each\ A_{net}^{j}\ in\ \mathbf{A_{net}^{0}}$ **do**
    $AM_j^t \leftarrow mutate\ AM_j\ by\ x\ Hamming\ distance$
    $LG_j^t \leftarrow mutate\ LG_j\ by\ x\ Hamming\ distance$
  end for
  for $i\ in\ 1:m$ **do**
    $initial\ system\ state \leftarrow I_{n,i}$
    $update\ f_0\ for\ each\ A_{net}\ and\ add\ penalty\ when\ f_0 \notin [0,1]$
    if $A_{net}\ has\ an\ independent\ self\ activating\ edge$ & $the\ initial\ system\ state\ in$
    $which\ the\ self\ activating\ gene\ has\ been\ set\ to\ its\ maximal\ expression \notin I_{n,m}$
    then
      $append\ the\ modified\ state\ described\ above\ to\ I_{n,m}$
    end if
    $initial\ system\ state \leftarrow initial\ system\ state\ + mild\ random\ perturbation$
    for $each\ A_{net}^{j}\ in\ \mathbf{A_{net}^{t}}$ **do**
      $integrate\ GRN\ ODEs\ with\ A_{net}^{j} : \{AM_j^t,\ LG_j^t,\ f_{oj}^t\}$
    end for
    $record\ last\_states_{i,j}$
    if $last\_states_{i,j}\ is\ a\ fixed\ point\ attractor$ **then**
      $record\ Attractor_{i,j}$
    else
      $add\ a\ penalty\ to\ A_{net}^{j}$
    end if
  end for
  $Calculate\ \min(AttractorDistance_{\mathbf{A_{net}^{t}}})$
  if $\min(AttractorDistance_{\mathbf{A_{net}^{t}}}) < \min(AttractorDistance_{\mathbf{A_{net}^{0}}})$ **then**
    $\mathbf{A_{net}^{0}} \leftarrow \mathbf{A_{net}^{t}}$
  else
    $\mathbf{A_{net}^{0}}\ remains$
  end if
  for $each\ A_{net}^{j}\ in\ \mathbf{A_{net}^{0}}$ **do**
    $fitness_{A_{net}^{j}} = fitness_{A_{net}^{j}} + \frac{1}{AttractorDistance_{A_{net}^{j}}}$
  end for
  $Sort\ \mathbf{A_{net}^{0}}\ by\ fitness\ in\ descending\ order : \{A_{net}^{(1)},\ A_{net}^{(2)},\ ...,\ A_{net}^{(N)}\}$
  $\mathbf{A_{net}^{0}} \leftarrow \{A_{net}^{(1)},\ A_{net}^{(2)},\ ...,\ A_{net}^{(0.8N)},\ A_{net}^{(1)},\ ...,\ A_{net}^{(1)}\}$
end for
$Return\ the\ A_{net}^{(1)}\ in\ \mathbf{A_{net}^{0}}$

Due to the fact that the sampling and mutating steps in the algorithm are stochastic, the output A_{net} can be different each time. We propose to draw a consensus GRN architecture with the assumption that a particular regulatory connection (i.e. an entry in A_{net}) occurring

at a high frequency among a group of inferred GRN architectures would be more important. In this case, we conduct 30 independent and identical GRN inference processes and obtain a consensus GRN architecture by recombining the most frequent connections.

In addition to regular transcriptional profiles, our model also can incorporate data from genetically engineered strains including knockouts and overexpressions. For instance, if the input transcription profiles were from knockout strains, we can set the $[R]$ of the inoperative genes to zero during the numerical integration. If an exogenous gene copy was imported to the biological system, we can add a constant to the derivative of the gene. If a regulatory connection was snapped by genetic engineering (e.g. disrupting a TF binding site in a promoter), we can fix the corresponding entry in A_{net} as zero to eliminate the effect of the disrupted regulatory connection. Furthermore, we can use chromatin immunoprecipitation (ChIP) data to guide the mutation of A_{net} during inference: if a physical binding interaction exists between a gene and a TF, they are likely to have a regulatory connection. Therefore, we can lower the probability of the corresponding entry being mutated to zero. The accommodation of different data types allows the model to integrate more available data and perform better.

Given a known GRN architecture and kinetic parameters, we have two straightforward approaches to explore the GRN state space searching for attractors. The first approach is called the global searching strategy, which generates evenly distributed starting points in the GRN state space and loops through them (Fig. 2.7a). Due to the high dimensions of GRN state spaces, this strategy is very inefficient and time consuming. Attractors in the gaps between searching areas could be missed. The second approach, called the local searching strategy, works much faster but it is only applicable when there is an expected attractor in an approximate location (Fig. 2.7b). This approach generates starting points around the expected attractor to examine the basin of attraction. If the systems were attracted to the anticipated spot and stabilized there, the attractor can be verified.

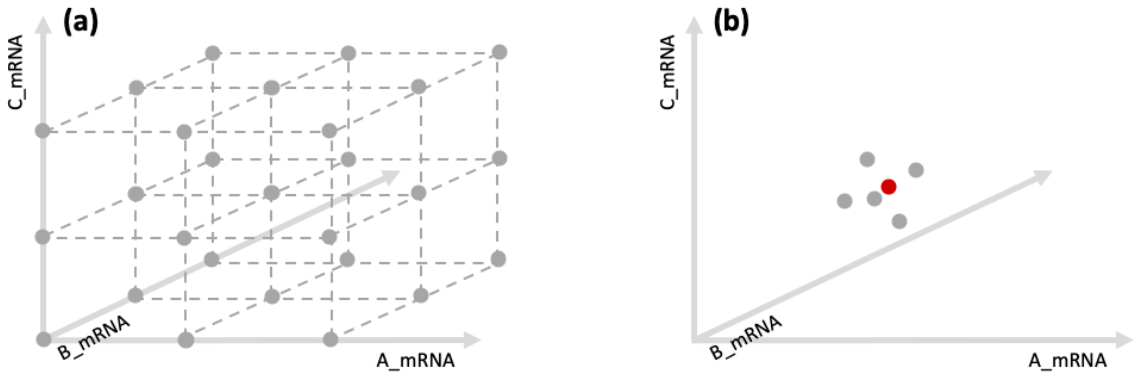


Figure 2.7: Two approaches to search for attractors given a known GRN system. (a) Global searching strategy: the GRN dynamic system will be initiated at each of the starting points (grey circles), which are evenly distributed in the state space. (b) Local searching strategy: starting points will be generated around an expected attractor (red circle). The GRN dynamic system must go from these starting points to the anticipated location to confirm the attractor.

Bibliography

[1] Blagoj Ristevski. Overview of computational approaches for inference of microrna-mediated and gene regulatory networks. In *Advances in Computers*, volume 97, pages 111–145. Elsevier, 2015.

[2] Tim Van den Bulcke, Koenraad Van Leemput, Bart Naudts, Piet van Remortel, Hongwu Ma, Alain Verschoren, Bart De Moor, and Kathleen Marchal. SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC bioinformatics*, 7(1):43, 2006. ISBN: 1471-2105 Publisher: Springer.

[3] Pedro Mendes, Wei Sha, and Keying Ye. Artificial gene networks for objective comparison of analysis algorithms. *Bioinformatics*, 19:ii122–ii129, 2003. ISBN: 1460-2059 Publisher: Oxford University Press.

[4] Thomas Schaffter, Daniel Marbach, and Dario Floreano. GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics*, 27(16):2263–2270, 2011. ISBN: 1460-2059 Publisher: Oxford University Press.

[5] Richard Bonneau, David J. Reiss, Paul Shannon, Marc Facciotti, Leroy Hood, Nitin S. Baliga, and Vesteinn Thorsson. The inferelator: an algorithm for learning

parsimonious regulatory networks from systems-biology data sets de novo. *Genome biology*, 7(5):R36, 2006. ISBN: 1474-760X Publisher: Springer.

[6] Barbara Di Camillo, Gianna Toffolo, and Claudio Cobelli. A gene network simulator to assess reverse engineering algorithms. *Annals of the New York Academy of Sciences*, 1158(1):125–142, 2009. ISBN: 0077-8923 Publisher: Wiley Online Library.

[7] Sui Huang, Gabriel Eichler, Yaneer Bar-Yam, and Donald E. Ingber. Cell fates as high-dimensional attractor states of a complex gene regulatory network. *Physical review letters*, 94(12):128701, 2005. Publisher: APS.

[8] Thuy Tien Bui and Kumar Selvarajoo. Attractor concepts to evaluate the transcriptome-wide dynamics guiding anaerobic to aerobic state transition in escherichia coli. *Scientific reports*, 10(1):1–14, 2020. ISBN: 2045-2322 Publisher: Nature Publishing Group.

[9] Tariq Enver, Martin Pera, Carsten Peterson, and Peter W. Andrews. Stem cell states, fates, and the rules of attraction. *Cell stem cell*, 4(5):387–397, 2009. ISBN: 1934-5909 Publisher: Elsevier.

[10] Akiko Kashiwagi, Itaru Urabe, Kunihiko Kaneko, and Tetsuya Yomo. Adaptive response of a gene network to environmental changes by fitness-induced attractor selection. *PloS one*, 1(1):e49, 2006. ISBN: 1932-6203 Publisher: Public Library of Science.

[11] Ryohei Morishita, Hiroaki Imade, Isao Ono, Norihiko Ono, and Masahiro Okamoto. Finding multiple solutions based on an evolutionary algorithm for inference of genetic networks by s-system. In *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, volume 1, pages 615–622. IEEE, 2003.

[12] Daisuke Tominaga, Nobuto Koga, and Masahiro Okamoto. Efficient numerical optimization algorithm based on genetic algorithm for inverse problem. In *Proceedings of the 2nd Annual Conference on Genetic and Evolutionary Computation*, pages 251–258, 2000.

[13] SR Paladugu, V Chickarmane, A Deckard, JP Frumkin, M McCormack, and HM Sauro. In silico evolution of functional modules in biochemical networks. *IEE Proceedings-Systems Biology*, 153(4):223–235, 2006.

[14] John Charles Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2016.

CHAPTER III

THEORETICAL VALIDATION

3.1 Model networks for validation

It is extremely challenging to directly determine the complete and comprehensive composition and structure of "real-world" GRNs in living organisms ([1, 2]). Therefore, the use of experimental data in GRN inference can be problematic when it comes to validating the outcome of GRN model predictions, since one can rarely if ever be certain that the experimental data provides a complete picture of the real-world GRN structure. For this reason, it has become common practice in the field of GRN inference to utilize *in silico* (i.e. computer generated) datasets for method validation, which can provide gene expression data that is directly predicted based on a hypothetical "source" GRN model ([3, 4, 5, 6]).

Both *in silico* and biologically observed GRN instances have been used to evaluate our approach. The *in silico* instance consists of five arbitrarily generated A_{net}, each of which has at least 9 different fixed-point attractors (Fig. 3.1). These A_{net} are regarded as the reference GRN architectures, which will be used as answer keys to examine the inferred GRN architectures. The *in silico* fixed-point attractors for each A_{net} are generated by SynTReN, a commonly used benchmark generator for GRN inference ([4]). Considering that noise generally exists in the experimentally-derived transcription profiles, we added a Gaussian distributed noise to the *in silico* input transcription profiles. The kinetic parameters of the model were assigned by the values experimentally measured in *E. coli* ([7, 8, 9, 10]) (See Table 3.1).

Table 3.1: Kinetic parameters used in *in silico* and real-life tests

Kinetic parameter	Derived values	Reference
mRNA elongation rate	4.8 nt./s	[7]
Ribosome elongation rate	8 aa./s	[9]
mRNA degradation rate	0.0067/s	[8]
Protein degradation rate	0.00796/s	[10]

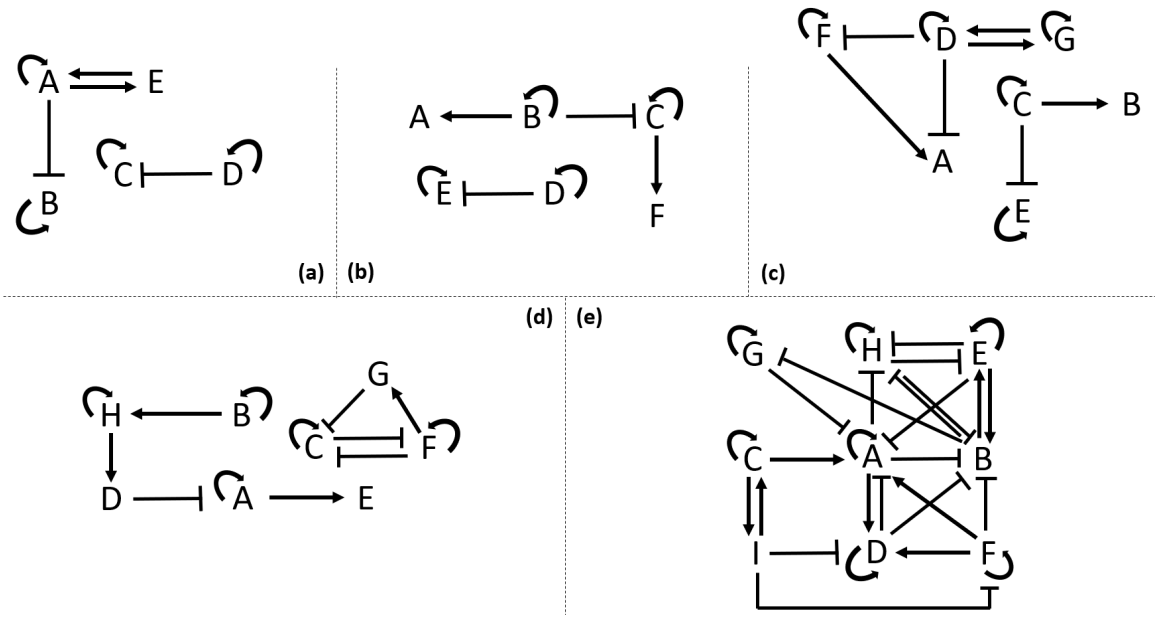


Figure 3.1: Five GRN architectures were arbitrarily generated as references in the *in silico* test. They have 5-9 (a-e) genes and at least 9 different fixed-point attractors. The pointed (or blunt) arrows represent activation (or repression) regulation.

3.2 Consensus GRNs converge upon attractors of reference GRNs

We used the attractors generated by the 5 *in silico* GRNs in Fig. 3.1 as input and performed 30 independent and identical inferences. When running the evolutionary algorithm, the attractor distance of the GRN instances gradually decreased, and eventually converged at a low level (Fig. 3.2). Although the velocities of convergence differ, none of the inferences was trapped in a local optimal in this case. We then drew a consensus GRN by selecting the most frequent edges in the adjacency matrices, fixing the logic gate

parameters, and taking the average of the f_0s. Since the consensus GRNs were inferred from the attractors of the reference GRNs, they should reproduce the same attractors in the input. We initiated a consensus GRN and a reference GRN randomly around their attractor position and the result showed that their expression levels converged upon the same attractor (Fig. 3.3).

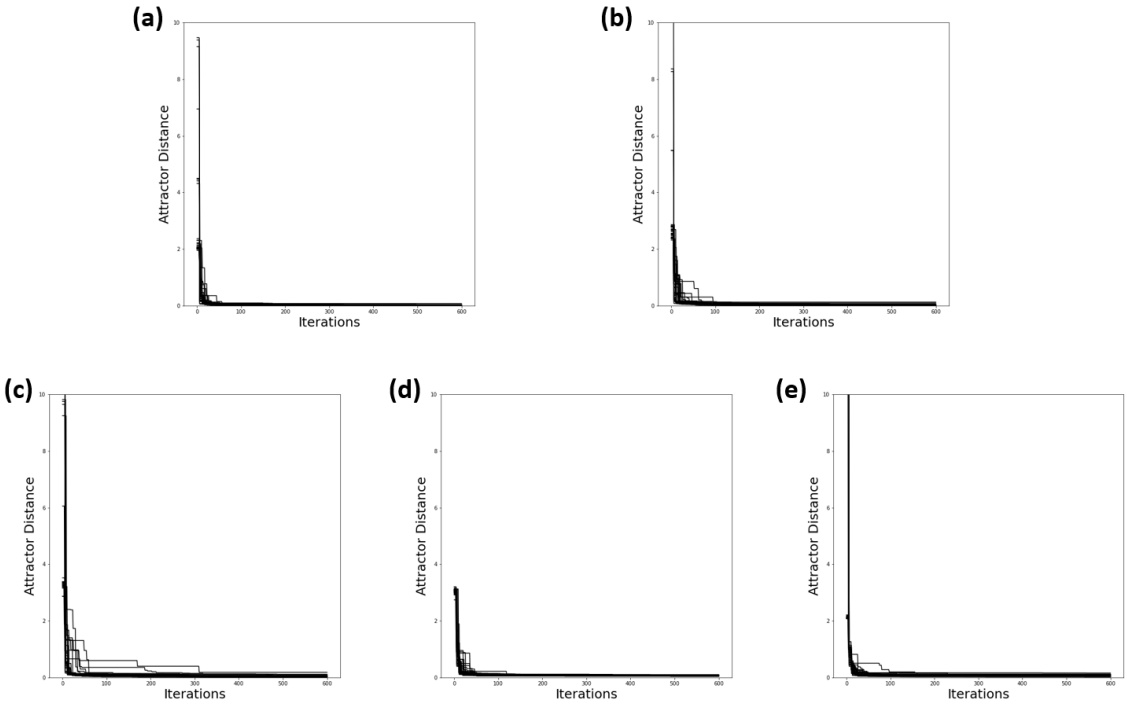


Figure 3.2: Convergence of GRN architecture on attractor distance. The x-axis represents the iteration numbers and the y-axis is the attractor distance between the input and the ones produced by the current GRN in training. Data obtained from the inference of the *in silico* GRNs. (a-e) 5-gene to 9-gene *in silico* GRNs. The evolutionary algorithm effectively searched for the GRNs that fit the input attractors in all 5 tests.

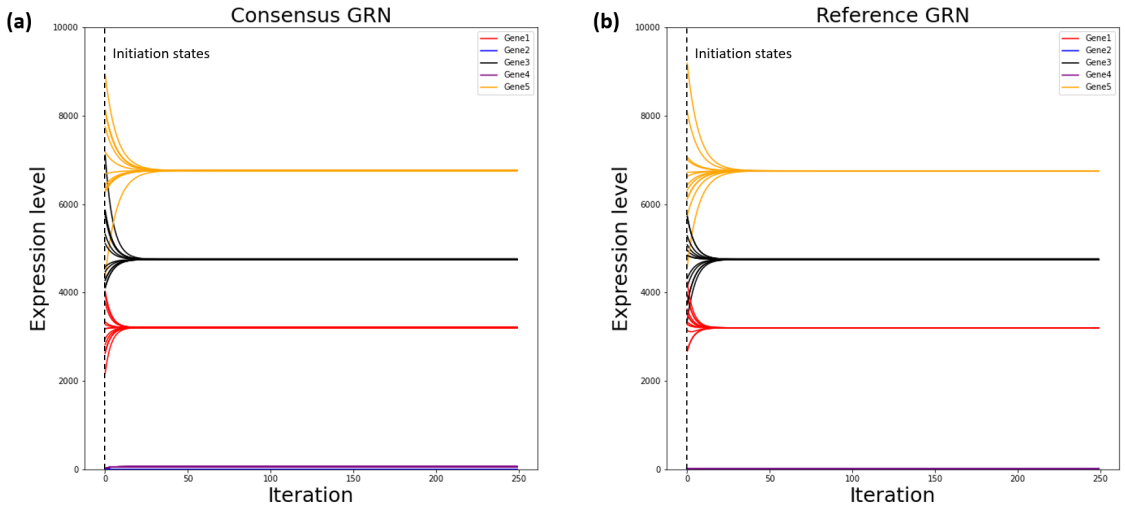


Figure 3.3: GRN dynamics when initiated around the attractor position. The consensus GRN (a) was inferred by the attractors of the 5-gene *in silico* reference GRN (b). The initial states were obtained by the attractor position plus a Gaussian distributed random variable.

3.3 GRN architecture and its attractor profiles are strongly coupled

One key hypothesis of our model is that a GRN architecture can be revealed by its attractors. In order to ensure that this hypothesis holds true under normal circumstances, we arbitrarily generated 5 *in silico* GRN architectures as test subjects (Fig. 3.1) to examine the correlation between their A_{net} and attractor profiles. Specifically, we randomly mutated the A_{net} of the 5 reference GRN architectures and observed how their attractor profiles change accordingly. The difference between the mutant GRN architecture ($A_{net}^{mut} = \{AM^{mut}, LG^{mut}\}$) and the reference ones ($A_{net}^{ref} = \{AM^{ref}, LG^{ref}\}$) is measured by the Hamming distance, and the difference between their attractor profiles is measured by the attractor distance given by Eq. II.15. As shown in Fig. 3.4, when the A_{net}^{mut} becomes more different from A_{net}^{ref}, its attractor profiles tend to be further away from the reference ones. This general trend between A_{net} and attractor profiles justifies the search strategy of our algorithm, whereby the A_{net}^{ref} is inferred by gradually mutating A_{net} and improving the distance between the population's attractor profiles and those of the reference GRN. Based on our observations, network mutations tend to be more efficient when considering the slopes between the Hamming distance and the attractor distance of these five *in silico* GRNs, whereas crossover operations appear to be more suitable for cases with smaller slopes. The significant fluctuation in the y-axis (attractor distance) was anticipated to have a minimal effect on the inference

process as the evolutionary algorithm gives precedence to identifying networks with smaller attractor distances. We performed sensitivity tests on key kinetic parameters and perturbed them by 50% to evaluate their impact on the correspondence between network structure and attractors. The results, depicted in Fig 3.5, showed that perturbations in V_{max} and D_{mRNA} had a more significant effect compared to V_{trl} and $D_{protein}$, potentially due to violations of the steady-state assumption. Consequently, we identified V_{max} and the attractors as essential inputs for the model, while other parameters can be estimated based on the steady-state assumption.

$$H(S^1, S^2) = \frac{1}{N} \sum_{i=1}^{N} |S_i^1 - S_i^2|. \tag{III.1}$$

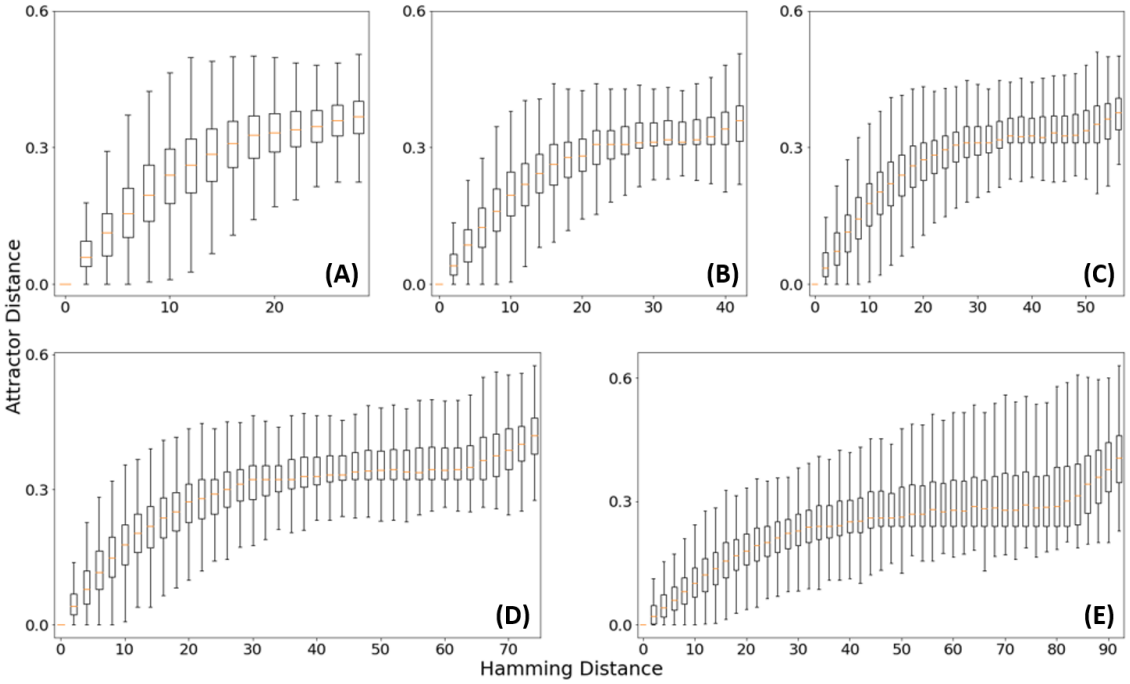


Figure 3.4: Positive correlation between A_{net} similarity (Hamming distance on x-axis) and attractor profiles similarity (attractor distance on y-axis). Each column in the box plots (A-E) contains 1000 random A_{net}^{mut} mutated from the 5 A_{net}^{ref} consisting of 5-9 genes.

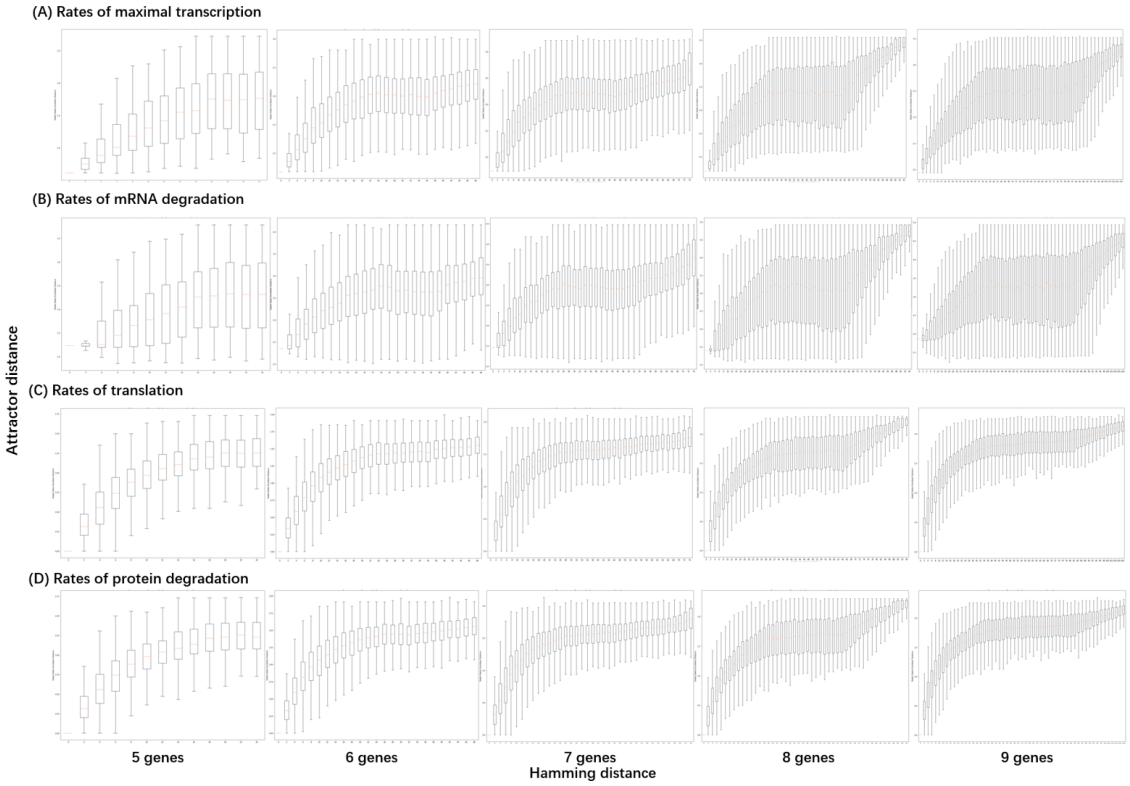


Figure 3.5: Sensitivity tests for the kinetic parameters, (A) rates of transcription, (B) mRNA degradation, (C) translation, and (D) protein degradation. Each of these parameters was perturbed by 50% of their original values and used to generate the correlation between A_{net} similarity (Hamming distance on the horizontal axis) and attractor profiles similarity (attractor distance on the vertical axis).

In addition, Fig. 3.4 shows that the attractor distance can reach zero before the Hamming distance goes to zero. It indicates that the same set of attractors can be generated by different GRNs, or in another word, more than one GRN architectures can satisfy the need for an organism to produce environment-fitting transcriptional profiles. We speculate that an organism's GRN can be mutated during DNA replication, since nucleotide mismatch can occur in TF coding genes and TF binding sites. As a result, multiple GRN architectures can simultaneously exist in a population of an organism.

3.4 Comparison against six other GRN inference methods on *in silico* test

We tested our model and other existing models using the 5 *in silico* reference GRNs. To avoid biasing the results in favor of our algorithm, we generated ODEs for these

five topologies using SynTReN ([4]). We generated simulated transcription profile inputs from the attractors of these ODEs. For each instance, we used the attractors as input and ran 600 iterations in algorithm 1. We performed 30 independent and identical inference processes and obtained a consensus A_{net} by weighted averaging edge frequencies. We compared the inferred consensus A_{net} to reference A_{net} in terms of F1 score, area under Receiver Operating Characteristic (ROC) and Precision and Recall (PR) curves, metrics that are commonly used in machine learning. We compared our model, named evolutionary algorithm (EA), to 6 other widely-used benchmark models, including ARACNE ([11]), CLR ([12]), GENIE3 ([13]), MRNET ([14]), MRNETB ([15]) and SIMONE ([16])). Amongst these models, only EA and GENIE3 can infer directed networks with asymmetric adjacency matrices (Table 3.2). Therefore, we symmetrized the inferred networks of EA and GENIE3 as EA_SYM and GENIE3_SYM by making all edges undirected. In addition, these models, except for EA, cannot infer the self-regulatory edges. The diagonal in their inferred adjacency matrix has only zeros. To bridge this discrepancy, we used the original reference GRNs to evaluate EA while treating all self-regulation as correctly identified for all other models; this disadvantages our algorithm in the benchmarks we present. Despite this, as presented by Fig. 3.6, the EA performed generally better than the other algorithms. The protein coordination parameters and f_0s of EA were also well converged on the ground truth (Table 3.3 and 3.4). Additionally, When the scale of the *in silico* GRN increases from 5 to 9 genes, it becomes harder to infer the AM. We believe that more attractor profiles are needed to reveal additional stable states of large-scale GRNs and to compensate the curse of dimensionality brought by its bigger state space volume.

We tested the ability of our model, and several other existing models, to infer GRN architectures using the attractor profiles generated by the 5 in silico reference GRNs depicted in Fig. 3. To avoid biasing the results in favor of our algorithm, we generated ODEs for these five topologies using SynTReN ([4]). We generated simulated transcription profiles from the attractors of these ODEs by global searching strategy and utilized them as the input for the in silico test. For each instance, we used the attractors as input and ran 600 iterations in algorithm 1. We performed 30 independent and identical inference processes and obtained a consensus Anet by weighted averaging edge frequencies. We compared the inferred consensus Anet to reference Anet using common machine learning metrics, including the F1 score, area under Receiver Operating Characteristic (ROC) and Precision and Recall (PR) curves. We compared our evolutionary algorithm (EA) method, to 6 widely used benchmark methods, including ARACNE ([11]), CLR ([12]), GENIE3 ([13]),

MRNET ([14]), MRNETB ([15]) and SIMONE ([16])). Amongst these methods, only EA and GENIE3 can infer directed networks with asymmetric adjacency matrices, which can differentiate the regulating gene and the target gene (Table 3.2). Therefore, we symmetrized the inferred networks of EA and GENIE3 as EA SYM and GENIE3 SYM by making all edges undirected. As presented by Fig. 3.6, the EA performed generally better than the other algorithms. The protein coordination parameters and f_0s of EA were also well converged on the ground truth ((Table 3.3 and 3.4)). With the exception of EA, none of the methods examined can infer self-regulatory edges; therefore the diagonal in their inferred adjacency matrix was set to zero. To bridge this discrepancy, we created another set of reference GRNs that contain no self-regulatory edges (Fig. S2). We conducted the same comparison on this set of GRNs and fixed the diagonal in inferred adjacency matrices as zeros. This disadvantages our algorithm in the benchmark we present because other methods are not capable of generating false positives for autoregulation. The result indicates that in the absence of auto-regulation, GENIE3 outperformed the other method and our method was among the most competent ones (Fig. S3). Additionally, when the scale of the *in silico* GRN increases from 5 to 9 genes, it becomes harder to infer the AM . We believe that more attractor profiles are needed to reveal additional stable states of large-scale GRNs and to compensate the curse of dimensionality brought by its bigger state space volume.

Table 3.2: Comparison of inference software features

Ability to infer	ARACNE	CLR	MRNET	MRNETB	SIMONE	GENIE3	EA
Directed GRN	No	No	No	No	No	Yes	Yes
Sign of regulation	No	No	No	No	No	No	Yes
Self-regulation	No	No	No	No	No	No	Yes
Characterization of protein-protein coordination	No	No	No	No	No	No	Yes

Table 3.3: The *in silico* test result for protein coordination matrix

in silico GRN instances	Hamming distance	Percentile	Accuracy	Precision	Recall
5-gene GRN	1.00(5.00)	1.08%(62.37%)	0.90(0.50)	1.00(0.82)	0.90(0.50)
6-gene GRN	3.00(6.00)	7.30 %(61.29%)	0.75(0.50)	1.00(1.00)	0.75(0.50)
7-gene GRN	3.00(7.00)	2.87%(60.47%)	0.79(0.50)	0.73(0.76)	0.79(0.50)
8-gene GRN	1.5(8.00)	0.12%(59.82%)	0.88(0.50)	0.88(0.88)	0.88(0.50)
9-gene GRN	4.00(9.00)	1.54%(59.27%)	0.78(0.50)	0.60(0.65)	0.78(0.50)

Values in parenthesis show the results of random LG. Accuracy, precision, and recall are calculated by weighted average (averaging the support-weighted mean per label).

Table 3.4: The *in silico* test result for $f0$

in silico GRN instances	Average $f0$	Std $f0$
5-gene GRN	[0.001, 0.016, 0.013, 0.042, 0.001]	8.28e-2
6-gene GRN	[0.023, 0.043, 0.018, 0.07, 0.004, 0.028]	1.42e-1
7-gene GRN	[0.039, 0.045, 0.046, 0.052, 0.045, 0.015, 0.019]	1.28e-1
8-gene GRN	[0.028, 0.029, 0.031, 0.228, 0.021, 0.036, 0.074, 0.3]	1.84e-1
9-gene GRN	[0.006, 0.021, 0.031, 0.27, 0.098, 0.054, 0.086, 0.095, 0.035]	1.14e-1

The f_0s are 0 for all *in silico* reference GRNs.

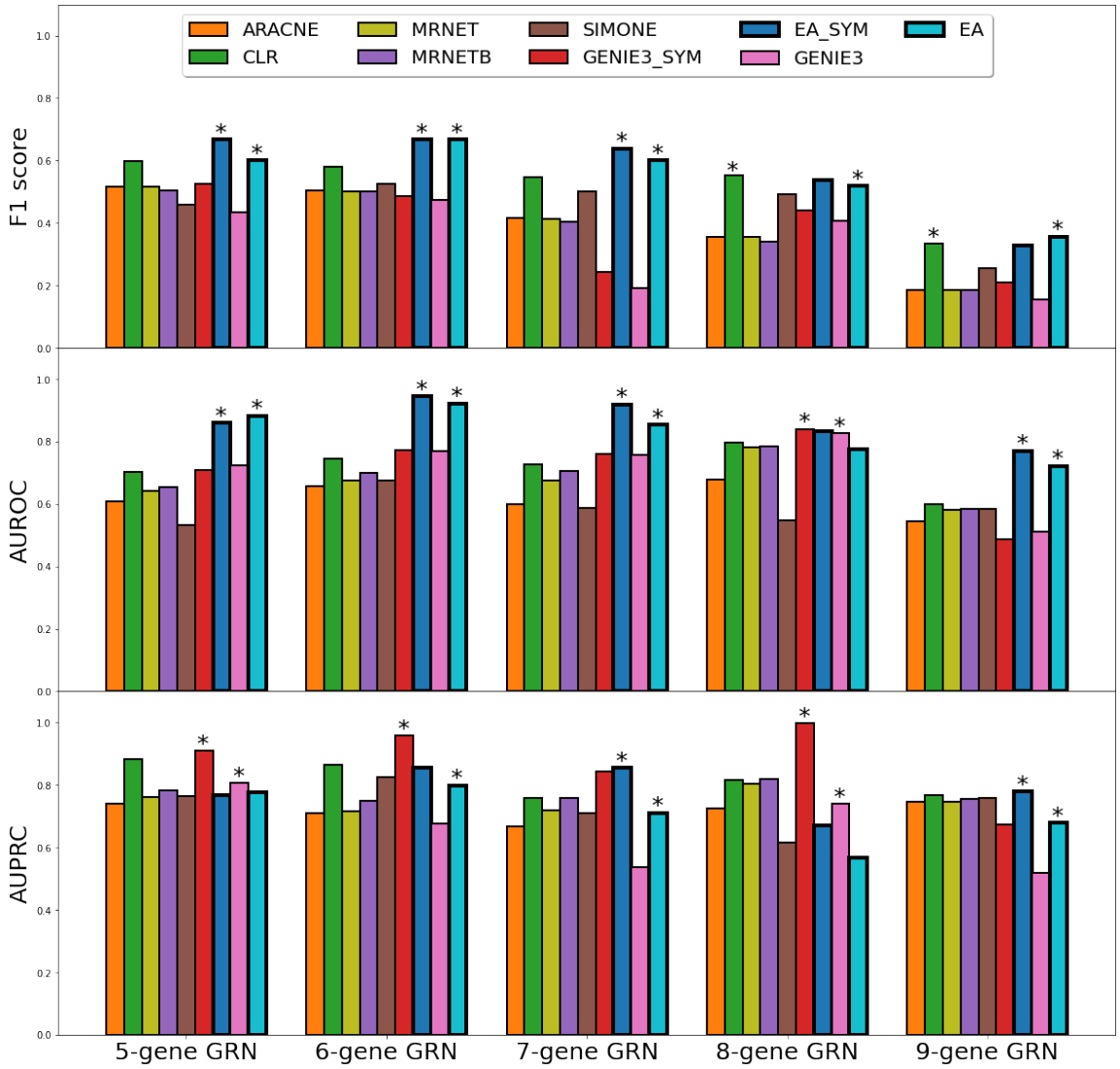


Figure 3.6: The *in silico* test comparison result in F1 score (upper panel), AUROC (middle panel), and AUPRC (bottom panel). The F1 scores are calculated using a threshold cutoff of 0.5 for all models. Best marked by a star for symmetric and asymmetric methods.

3.5 Our model can predict novel attractors produced by single-knockout GRNs

Since the inferred GRNs are similar to the true GRNs on both structure and capability to produce attractors, we anticipated that the inferred GRNs should be able to predict the dynamics of the true ones even when they are mutated. To examine this ability, we knocked out the genes one by one in the five *in silico* reference GRNs and found their new attractors by global searching strategy. These new attractors are unknown

to the inferred GRNs because they had not been used in the GRN inference process. We performed the same knockouts in the inferred GRNs to see if they could accurately predict the new attractors of the mutated reference GRNs. The attractors of the mutated reference GRNs were generated by SynTReN, while the attractors of the mutated inferred GRNs were predicted by our model. We applied systematic global searching to find attractors and used a random GRN as control. We found that the single-knockout reference GRNs had altogether 384 attractors (combining attractors from all knockouts across all five reference GRNs) and the single-knockout inferred GRNs had a combined total of 385 attractors. Of these attractors, 273 (71.1% of the reference GRN attractors and 70.9% of the inferred GRN attractors) were matched. The random GRN showed 32 attractors and none was matched (Table 3.5). See full report in the supplemental material.

Table 3.5: Attractor prediction result summary by global searching strategy

in silico GRN instances	Attractors in single-knockout (reference)	Attractors in single-knockout (inferred)	Matched attractors
5-gene GRN	33	36	33
6-gene GRN	42	61	42
7-gene GRN	57	60	54
8-gene GRN	88	116	79
9-gene GRN	164	112	65
Total number	384	385	273

Values in parenthesis show the results of random a GRN. Two attractors considered matched have an attractor distance less than 0.15. No attractors were matched in a random GRN.

From this result, we can conclude that the attractors generated by a GRN are tied together to an extend. If a subset of the attractors were successfully reproduced by the inferred GRN, it is likely that more attractors can be discovered using the inferred GRN. The success rate of such "attractor discovery" can indicate how close the inferred GRN is to the real one. Under the real-life biological condition in which we cannot tell the full picture of the *in vivo* GRN by experiments, it would be a considerable approach to estimate how well the *in vivo* GRN is perceived.

Bibliography

[1] Maria I. Arnone and Eric H. Davidson. The hardwiring of development: organization and function of genomic regulatory systems. *Development*, 124(10):1851–1864,

1997. ISBN: 0950-1991 Publisher: The Company of Biologists Ltd.

[2] Blagoj Ristevski. Overview of computational approaches for inference of microrna-mediated and gene regulatory networks. In *Advances in Computers*, volume 97, pages 111–145. Elsevier, 2015.

[3] Thomas Schaffter, Daniel Marbach, and Dario Floreano. GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics*, 27(16):2263–2270, 2011. ISBN: 1460-2059 Publisher: Oxford University Press.

[4] Tim Van den Bulcke, Koenraad Van Leemput, Bart Naudts, Piet van Remortel, Hongwu Ma, Alain Verschoren, Bart De Moor, and Kathleen Marchal. SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC bioinformatics*, 7(1):43, 2006. ISBN: 1471-2105 Publisher: Springer.

[5] Pedro Mendes, Wei Sha, and Keying Ye. Artificial gene networks for objective comparison of analysis algorithms. *Bioinformatics*, 19:ii122–ii129, 2003. ISBN: 1460-2059 Publisher: Oxford University Press.

[6] Diogo Camacho, PAOLA VERA LICONA, Pedro Mendes, and Reinhard Laubenbacher. Comparison of reverse-engineering methods using an in silico network. *Annals of the New York Academy of Sciences*, 1115(1):73–89, 2007. ISBN: 0077-8923 Publisher: Wiley Online Library.

[7] Stefan Klumpp and Terence Hwa. Growth-rate-dependent partitioning of RNA polymerases in bacteria. *Proceedings of the National Academy of Sciences*, 105(51):20245–20250, 2008. ISBN: 0027-8424 Publisher: National Acad Sciences.

[8] Mary Ann Moran, Brandon Satinsky, Scott M Gifford, Haiwei Luo, Adam Rivers, Leong-Keat Chan, Jun Meng, Bryndan P Durham, Chen Shen, Vanessa A Varaljay, et al. Sizing up metatranscriptomics. *The ISME journal*, 7(2):237–243, 2013.

[9] Călin C Guet, Luke Bruneaux, Taejin L Min, Dan Siegal-Gaskins, Israel Figueroa, Thierry Emonet, and Philippe Cluzel. Minimally invasive determination of mrna concentration in single living bacteria. *Nucleic acids research*, 36(12):e73–e73, 2008.

[10] MR Maurizi. Proteases and protein degradation inescherichia coli. *Experientia*, 48(2):178–201, 1992.

[11] Adam A Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. In *BMC bioinformatics*, volume 7, pages 1–15. BioMed Central, 2006.

[12] Jeremiah J. Faith, Boris Hayete, Joshua T. Thaden, Ilaria Mogno, Jamey Wierzbowski, Guillaume Cottarel, Simon Kasif, James J. Collins, and Timothy S. Gardner. Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS biol*, 5(1):e8, 2007. ISBN: 1545-7885 Publisher: Public Library of Science.

[13] Vân Anh Huynh-Thu, Alexandre Irrthum, Louis Wehenkel, and Pierre Geurts. Inferring regulatory networks from expression data using tree-based methods. *PloS one*, 5(9):1–10, 2010.

[14] Patrick E Meyer, Kevin Kontos, Frederic Lafitte, and Gianluca Bontempi. Information-theoretic inference of large transcriptional regulatory networks. *EURASIP journal on bioinformatics and systems biology*, 2007:1–9, 2007.

[15] Patrick Meyer, Daniel Marbach, Sushmita Roy, and Manolis Kellis. Information-theoretic inference of gene networks using backward elimination. In *BioComp*, pages 700–705, 2010.

[16] Julien Chiquet, Alexander Smith, Gilles Grasseau, Catherine Matias, and Christophe Ambroise. Simone: Statistical inference for modular networks. *Bioinformatics*, 25(3):417–418, 2009.

CHAPTER IV

APPLICATIONS ON *IN VIVO* REGULATORY NETWORKS

4.1 Materials used in *in vivo* tests

To test our model against an *in vivo* GRN instance, we used experimental data derived from a synthetic GRN engineered in *Saccharomyces cerevisiae* by Cantone et al. ([1]). Consisting of 5 genes and a variety of regulatory interactions (Fig. 4.1 a), the GRN can switch amongst 10 distinct stable states in response to the overexpression of each individual gene in two different carbon sources, galactose and glucose. These stable states were measured by quantitative PCR (qPCR) and converted to absolute expression levels. The promoter strengths, which indicate the rates of transcription initiation for each gene in the GRN, have been estimated by a stochastic optimization algorithm from steady-state gene expression data measured by qPCR ([1]). Other kinetic parameters used in the *in vivo* test are provided by Table 3.1. The A_{net}, transcription profiles, and kinetic parameters for the *in silico* and *in vivo* tests can be found in the supplemental material.

We also tested our model using transcriptional profiles derived from a set of 12 wild-type and targeted TF deletion strains of *Candida albicans*. All strains used in this study are described in Table S3 and are derived from SN156, which is a commonly used derivative of the SC5314 strain that is used widely in *C. albicans* studies ([2], [3]). All of the *C. albicans* single TF deletion strains used in this study were reported previously ([3]). TF double deletion strains were generated using CRISPR-mediated genome editing to delete the *WOR1* coding sequence as described by Nguyen et al ([4]). Steady-state transcript levels were determined using the 3' quant seq RNA sequencing methodology as described by Moll et al ([5]). Briefly, *C. albicans* cells were harvested from mid to late-log cultures and total RNA was isolated using the RiboPure™ RNA Purification kit. cDNA

libraries were prepared using the QuantSeq 3' mRNA-Seq Library Prep kit from Lexogen and multiplexed in pools of 96 libraries. Single-end 100bp reads were obtained using an Illumina HiSeq4000 instrument. The resulting de-multiplexed sequencing reads were trimmed and aligned using STAR Aligner ([6]) to obtain raw read counts for each transcript genome-wide. The promoter strengths of each gene in the network were determined using capped small RNA (csRNA) sequencing ([7]). This method enables the isolation of short nascent mRNA transcripts, rather than full-length mRNAs, and thus provides an instantaneous snapshot of the level of transcriptional activity at each transcriptional start site, genome-wide. Briefly, we enriched for nascent small, capped, RNA molecules from total RNA extracted from mid-log phase *C. albicans* cultures and prepared sequencing ready libraries using the small RNA library preparation kit from New England Biolabs. The resulting libraries were multiplexed and 16 indexed libraries were pooled prior to sequencing on an Illumina HiSeq4000 instrument. Sequencing data were analyzed using HOMER ([7]). The mRNA and csRNA sequencing data can be accessed on GEO (GSE217461 and GSE217383), and our algorithm is available at GitHub (`https://github.com/UCM-RuihaoLi/GeneRegulatoryNetworkInference.git`).

To account for noise in the experimentally derived transcriptional profiles we measured the "average replicate distance" which describes the average pairwise attractor distance between each of the three biological replicates for each genotypic/phenotypic combination. This metric was then used to determine whether a given GRN model prediction was considered successful, with the basic premise that a successful GRN prediction should yield a transcriptional profile that lies within the average noise range of 60% (Table 4.2 and 4.3) in the experimentally derived transcriptional profiles. Since some of the experimental replicates had high variance, we also included an attractor distance threshold of 0.16. This threshold was selected based on the performance of a null model, which samples from a uniform distribution with the upper and lower limits as the maximal and minimal expression levels for each gene. For transcriptional profiles with five genes or more, the null model has a 5% chance or less to generate a profile below this cutoff of 0.16 (See Table S2).

4.2 Our algorithm revealed unintended edges in an engineered *S. cerevisiae* GRN

To examine how well the GRN dynamic model produced by our algorithm simulates experimentally derived gene expression and to what extent it is robust to measurement

noise, we tested our approach using experimental data derived from an engineered synthetic GRN in *S. cerevisiae* ([1]). This engineered GRN consists of seven activating or inhibitory edges and five genes, some of which are under control of non-native promoters (Fig. 4.1 a). Using the experimentally measured promoter strengths and ten distinct steady state gene expression profiles, derived from strains which individually overexpress each of the five genes in galactose and glucose, our model inferred the GRN shown in Fig. 4.1 b. In glucose, Gal80 blocks Gal4 from activating *SWI5*, while galactose can inactivate Gal80 and Gal4 is free to activate *SWI5*.

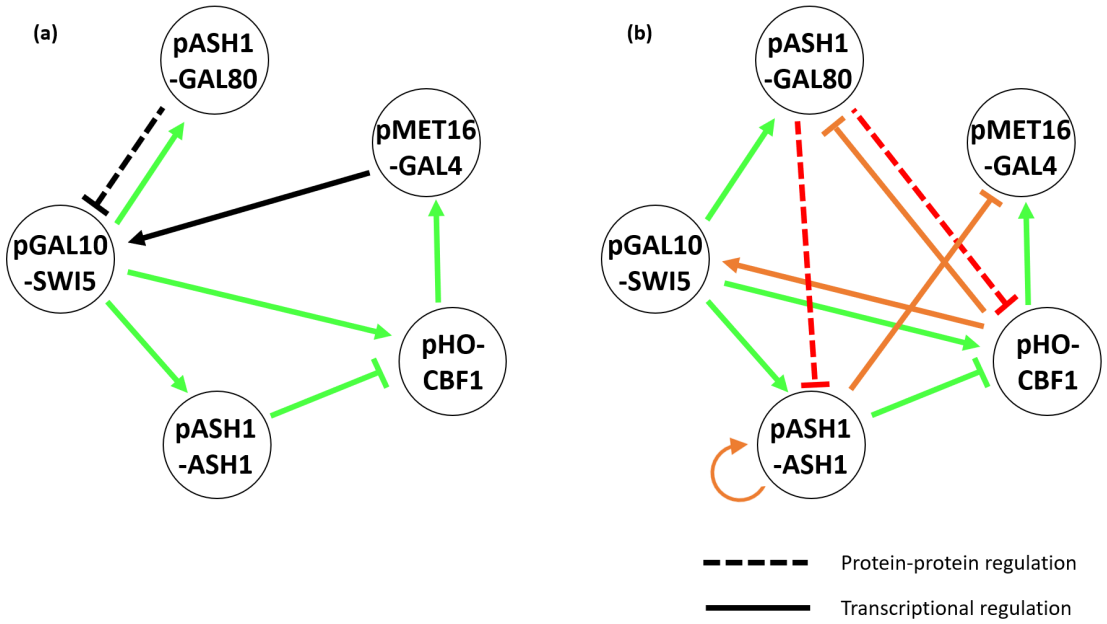


Figure 4.1: (a) The schematic diagram of the *S. cerevisiae* synthetic circuit. Solid lines represent direct transcriptional regulation and dotted lines indicate indirect transcriptional regulation mediated by a protein-level activation or inhibition of a transcription factor. Gal80 protein can inhibit *SWI5* transcription by preventing Gal4-mediated activation of target genes in the absence of galactose. Modified from the original paper ([1]). (b) The schematic diagram of the inferred circuit. Correctly inferred edges are labeled in green; additional edges not present in the original design of the circuit are labeled in orange and have support from the literature; edges labeled in red accurately describe the protein-level inhibitory effect of Gal80 on *SWI5*, as described in the text.

Our algorithm correctly identified five of the six transcriptional regulatory edges present in the original design of the engineered GRN (see comparison result in the supplemental material). In addition, our algorithm predicted two additional edges related to protein-protein interactions and four that were not intended in the original design of the engineered GRN, but for which there is experimental evidence in the literature (see Table 4.1). We

believe the missing transcriptional regulation of *SWI5* by Gal4 can be explained as follows. First, as shown in Fig. 3.4, the same set of attractors can be produced by different GRNs. In this case, the activation of *SWI5* by a feedback loop via *CBF1* and *GAL4* is replaced by a more direct activation by *CBF1* only. Second, the difference in the regulatory effects of Gal80 is related to how protein-protein interactions are encoded in our ODE framework. Special care must be taken in interpreting protein-protein interactions in the context of the inferred network produced by our algorithm. Our algorithm does not incorporate explicit protein-protein interactions, such as the interaction between Gal80 and Gal4, which leads to the downregulation of *SWI5* and furthermore *ASH1* and *CBF1*. Thus, in our inferred network, the inhibitory edge from *GAL80* to *SWI5* is not present. Instead, this protein inhibition is incorporated into the regulatory function for the targets of Swi5. Specifically, the Swi5 protein activates *CBF1* and *ASH1* transcription, but the protein Gal80 interferes with this activation. Therefore, at the mRNA level, increased *GAL80* transcription does not directly decrease *SWI5* mRNA production; rather it decreases *ASH1* and *CBF1* transcription. Thus, the inhibitory effect of Gal80 on the Swi5 protein is represented as the two inhibitory edges from *GAL80* to *ASH1* and *CBF1*. With this consideration in mind, only the self-activation of *ASH1*, the inhibition of *GAL4* by *ASH1*, the activation of *SWI5* by *CBF1*, and the inhibition of *GAL80* by *CBF1* represent regulatory effects that are not present in the intended synthetic system. We found previous experimental evidence for all these interactions in the literature (see Table 4.1).

Furthermore, while investigating the source of these additional edges, we observed that certain elements of the experimentally derived transcriptional profiles did not appear to be consistent with the intended design of the engineered GRN as described by Cantone et al. ([1]). Specifically, Cbf1 was intended to serve as the sole activator of *GAL4*, which in turn was meant to serve as the sole activator of *SWI5*. This would imply that, at steady state, *SWI5* should be expressed if and only if Cbf1 is elevated and *GAL4* is expressed. The experimentally derived transcriptional profiles contradict this. They indicated that at steady state, *SWI5* was activated even when *GAL4* was not expressed. Cantone et al. argued that *GAL4* is transiently expressed during an early phase of the experimental protocol, and that the Gal4 protein could persist to activate *SWI5* even after *GAL4* mRNA levels drop. However, this argument contradicts the steady state assumption of the transcriptional data and furthermore does not explain why *GAL4* mRNA levels were low when *CBF1*, which was intended to activate *GAL4*, was overexpressed.

We speculate that these discrepancies between the intended engineered GRN and the experimentally derived data may be explained by unintended regulatory interactions that modify the GRN structure and dynamics. By performing a systematic search on each TF-promoter pair in the intended engineered GRN using YEASTRACT+ ([8]), we uncovered support for this hypothesis. Specifically, we found experimental evidence from microarray, Northern blot, ChIP, and electrophoretic mobility shift assay (EMSA) experiments, supporting the idea that Cbf1 and Ash1 proteins regulate more than their intended target genes in the circuit. In fact, all four promoters within the circuit can be responsive to Cbf1 and Ash1 (Table 4.1).

Table 4.1: Experimental evidence for regulatory associations in the synthetic circuit

	Cbf1	Ash1	Gal4	Gal80	Swi5
pHO_CBF1	Inhibition [9]	Inhibition [10, 11]			Activation [12, 13]
pGAL10_SWI5	Activation [9]	[8]	Activation [14, 15]	Inhibition [16]	
pMET16_GAL4	Activation [17, 9]	[8]			
pASH1_GAL80	Activation [9]	[8]			Activation [10]
pASH1_ASH1	Activation [9]	[8]			Activation [10]

Column names are the TF proteins and row names are the promoters followed by their open reading frames. Orange shaded boxes indicate potential regulatory associations found in YEASTRACT+; blue shaded boxes are experimental evidence found by microarray and/or Northern blot experiments; pink shaded boxes are experimental evidence found by both microarray/Northern blot and ChIP/EMSA experiments.

Our inferred GRN predicted additional regulatory interactions beyond those that were intended in the synthetic regulatory network ([1]), and we identified experimental support for these putative regulatory interactions (Table 4.1). We conclude that the inferred GRN may have identified actual regulatory associations that impacted the experimentally derived transcriptional profiles, thus allowing our inferred GRN to accurately reproduce the experimentally measured attractor states and resolve the conflict between the intended GRN and the experimentally derived transcriptional profiles. This conclusion is supported by the observation that the attractors reproduced by our inferred GRN have 25.8% of the attractor distance of the mathematical model built by Cantone et al. (supplemental material). Furthermore, the experimental transcriptional profiles showed that *SWI5* was

repressed during overexpression of *GAL80* in both galactose and glucose, which was inconsistent with the intended GRN. The attractors produced by our model showed a consistent result: *SWI5* was suppressed when *GAL80* was overexpressed in glucose media, and it was expressed when galactose inactivated Gal80. Our model also explains the low expression of *GAL4* under *CBF1* overexpression: when *CBF1* was overexpressed, *ASH1* was activated by Cbf1 by two feed-forward loops (one via *SWI5* and the other via *GAL80*), and Ash1 in turn inhibited *GAL4*, lowering its expression. Together these results strongly suggest that our evolutionary algorithm approach to model construction can provide significant insight into the structure and regulatory dynamics of "real world" *in vivo* GRNs.

4.3 Modeling the white-opaque switch GRN in *C. albicans*

To expand beyond our model testing using data derived from "known" *in silico* and engineered *in vivo* GRNs, we next applied our algorithm to infer and simulate the dynamics of a naturally occurring GRN which controls reversible differentiation between two distinct cell types—white and opaque—in the human fungal pathogen *C. albicans*. The white and opaque cell types are heritably maintained for hundreds of generations and the frequency of stochastic switching between these two cell types is controlled by a complex, highly interwoven series of transcriptional regulatory interactions ([2]). The white and opaque cell types differ in the expression of approximately 18% of all genes in the *C. albicans* genome, thus providing two very distinct attractor states for the underlying GRN. To model the white-opaque GRN, we utilized transcriptional profiles derived from wildtype white and opaque cells, along with a series of strains that lack one or more of the TFs controling the switch (See Table S3). These additional TF deletion strains serve to provide additional steady-state attractors to further constrain the GRN structures. The majority of these strains can switch reversibly between the white and opaque cell types, thus providing two distinct attractor states per strain, with the exception of those TF deletion strains that are "locked" in one cell type or the other. In total, we obtained RNAseq data for seventeen distinct genotypic/phenotypic combinations including two wildtype strains, thirteen single TF deletion strains, and two double TF deletion strains. Each of the deleted TFs is known to impact the frequency of switching between the white and opaque cell types, and is known or predicted to impact the transcriptional profile of the resulting white and/or opaque cell types.

We first tested the ability of our evolutionary algorithm to predict the "unknown" transcriptional profiles produced by the GRNs of the wildtype and single TF deletion strains

by omitting the transcriptional profile(s) of a specific genotype from the training dataset and allowing the model to predict the omitted transcriptional profile(s). Transcriptional profiles from the two double TF deletion strains were excluded from all training sets and were reserved as final test subjects for a "fully trained" version of the model developed using the full complement of fifteen wildtype and single TF deletion strain transcriptional profiles as the training dataset. If the attractor distance between the predicted and omitted transcriptional profile(s) was below the average replicate distance, or a cutoff of 0.16, the prediction would be considered successful. The cutoff of 0.16 was selected by the null model, which has less than a 2% chance of generating a result below this cutoff for eight and nine gene networks (See Table S2). Since the null model produces transcriptional profiles by simply picking a value between the maximal and minimal expression levels, while the GRN dynamic system generates transcriptional profiles by numerically solving the differential equations, potential discrepancies may exist between the two. To rule out potential discrepancies due to the GRN dynamic system, we also generated 10,000 random GRNs as a control group and performed the same predictions on the omitted transcriptional profiles. Generally, half of the random GRNs produced fixed-point attractors, while the other half did not reach a steady state. Both the null model and the control GRNs showed similar distribution on their attractor distances and had an average of approximately 0.3.

Overall, nine out of the fifteen omitted wildtype and single TF deletion strain transcriptional profiles were successfully predicted by our model (Table 4.2). Of these nine successful predictions, eight had an average attractor distance of less than 0.16, and the last one (Δ/Δ*efg1*; Table 4.2) had an attractor distance above 0.16 but below the average replicate distance, meaning that the predicted transcriptional profiles were within the range of noise in the experimentally derived transcriptional profiles for the *EFG1* deletion strain. The six remaining prediction results showed either attractors exceeding the cutoff, or no attractor at all (indicated by dashes). We note that several of the experimentally derived transcriptional profiles had unusually high variability, as indicated by high average replicate distance values (Δ/Δ*wor3* opaque, Δ/Δ*czf1* opaque, and Δ/Δ*rbf1* opaque; Table 4.2). This high variability suggests excessive noise in the RNAseq libraries, or multiple states/oscillations existing in these specific cell types, either of which would violate the model assumption of a single stable-state transcriptional profile and make it challenging to evaluate the prediction. If we exclude these highly variable samples, the success rate of the model predictions increases to 66.7%.

Table 4.2: *C. albicans* wildtype and single TF deletion strains transcriptional profiles prediction results

Genotypes	Phenotype	Prediction trial 1	Prediction trial 2	Average replicate distance (noise range)	Control
Wildtype and single TF deletion strains					
wildtype	White	0.087	0.084	0.130 (42%)	0.329(57%)
wildtype	Opaque	–	0.506	0.231 (107%)	0.352(59%)
Δ/Δ*wor1*	White	0.119	0.131	0.156 (58%)	0.281(50%)
Δ/Δ*wor2*	White	0.114	0.093	0.252 (89%)	0.295(60%)
Δ/Δ*wor3*	White	0.146	0.144	0.152 (49%)	0.310(47%)
Δ/Δ*wor3*	Opaque	–	0.319	0.463 (144%)	0.286(54%)
Δ/Δ*wor4*	White	0.128	0.120	0.107 (42%)	0.310(53%)
Δ/Δ*efg1*	White	0.198	0.182	0.227 (78%)	0.314(56%)
Δ/Δ*efg1*	Opaque	–	–	0.120 (42%)	0.303(57%)
Δ/Δ*ahr1*	White	0.153	0.164	0.229 (72%)	0.319(62%)
Δ/Δ*ahr1*	Opaque	0.269	0.254	0.204 (94%)	0.279(55%)
Δ/Δ*czf1*	White	0.150	0.166	0.237 (81%)	0.340(45%)
Δ/Δ*czf1*	Opaque	0.327	0.353	0.282 (100%)	0.287(60%)
Δ/Δ*ssn6*	Opaque	0.254	0.366	0.182 (89%)	0.314(98%)
Δ/Δ*rbf1*	Opaque	0.136	0.139	0.334 (168%)	0.295(46%)

The transcriptional profile(s) of a specific genotype was left out in each prediction. Predictions whose attractor distances are no greater than 0.16 or the average replicates distance of the experimental data are indicated by green shaded boxes. Predictions that show no attractor (represented by dashes) or attractors exceeding the cutoff are indicated by unshaded boxes. Highly variable samples are indicated by gray shaded boxes. For the control, the average attractor distances of random GRNs that produced fixed-point attractors are indicated by the decimal values, while the percentage of random GRNs that produced fixed-point attractors is indicated in parentheses.

Next, we applied all fifteen of the wildtype and single TF deletion strain transcriptional profiles as training data to infer a consensus "fully trained" GRN. This consensus fully trained GRN was derived from thirty inferred GRN architectures and then used to predict the transcriptional profiles for two distinct double TF deletion strains. Since more attractors were used in the input, we anticipated that this consensus fully trained GRN should have a higher predictive power than the partially trained model. Both double TF deletion predictions were successful (Table 4.3), indicating that the transcriptional profiles produced by the consensus fully trained GRN closely mirror the experimentally derived transcriptional profiles for these two strains. Given the predictive accuracy of the consensus fully trained GRN, we next asked whether the underlying architecture, or adjacency matrix, of the inferred

GRN also closely resembled the experimentally determined binding interactions between these regulators and their respective coding genes, as previously reported ([2, 18, 19]). The GRN architectures inferred by the fully trained model are relatively diverse, with an average success rate of approximately 50% in predicting the experimentally determined TF-gene binding interactions observed in the ChIP data (Fig. 4.2). This discrepancy is not entirely unexpected given that our *in silico* testing demonstrated that multiple distinct GRN structures, covering a wide range of hamming distances, are capable of producing virtually identical transcriptional profiles, or attractor distances (Fig. 3.4).

Table 4.3: *C. albicans* double TF deletion transcriptional profiles prediction results

Genotypes	Phenotype	Prediction trial 1	Prediction trial 2	Average replicate distance (noise range)	Control
Δ/Δ*ssn6* Δ/Δ*wor1*	White	0.094	0.097	0.051 (20%)	0.263(99%)
Δ/Δ*rbf1* Δ/Δ*wor1*	White	0.154	0.155	0.118 (49%)	0.282(54%)

The transcriptional profile(s) of a specific genotype was left out in each prediction. Predictions whose attractor distances are no greater than 0.16 or the average replicate distance of the experimental data are indicated by green shaded boxes. Predictions that show no attractor (represented by dashes) or attractors exceeding the cutoff are indicated by unshaded boxes. Highly variable samples are indicated by gray shaded boxes (none present for these results). For the control, the average attractor distances of random GRNs that produced fixed-point attractors are indicated by the decimal values, while the percentage of random GRNs that produced fixed-point attractors is indicated in parentheses.

Given the enormous number of potential GRN architectures in the search space, and the fact that distinct GRNs, which produce identical attractors cannot be differentiated based purely on transcriptional profiles, we asked whether incorporating TF binding constraints could enable the model to converge upon an architecture that more closely resembles the experimental ChIP data while simultaneously reproducing accurate transcriptional profiles. To bias the model toward the GRN architecture observed in the experimental data, we included a TF binding probability function in our evolutionary algorithm. Briefly, this function alters the probability of an edge being created or removed in the adjacency matrix, thus biasing the inferred GRNs towards the experimentally determined architecture. However, if the resulting GRNs fail to converge upon the experimental attractors, the evolutionary algorithm would ultimately converge upon a distinct GRN structure if needed to fit the

transcriptional profiling data. We applied all seventeen of the transcriptional profiles used above, plus the previously published *in vivo* TF-DNA binding data, to infer "directed" GRNs. On average, the individual directed GRNs retained approximately 90% of the experimentally determined TF binding interactions while also reproducing most of the experimentally derived transcriptional profiles (Fig. 4.2). The consensus directed GRN, constructed by the high-frequency edges of the individual directed GRNs, accurately reproduced thirteen out of the seventeen experimentally observed transcriptional profiles and eighty out of the eighty-one physical binding interactions between each of the regulatory TFs and their respective coding genes. The transcriptional profiles that the consensus directed GRN failed to incorporate were wildtype opaque, Δ/Δ*wor3* opaque, Δ/Δ*ahr1* opaque, and Δ/Δ*ssn6* opaque, most of which had relatively high variability in their biological replicates (see full report for both *in silico* and *in vivo* prediction tests and inferred GRNs in the supplemental material). Together these results indicate that it is indeed possible to converge upon a GRN structure that closely mirrors the experimentally determined TF-DNA binding data for the white-opaque switch, while accurately producing many of the same attractor states observed *via* RNAseq. However, this data also suggests a high degree of redundancy or potential for plasticity within the white-opaque GRN, thus compromising the ability of our model to infer the observed GRN structure based solely on transcriptional profiling data.

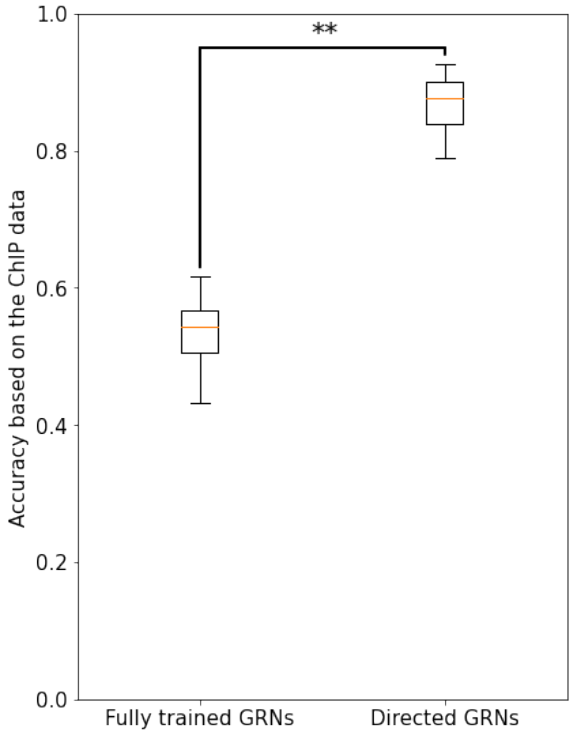


Figure 4.2: Accuracy distributions of the fully trained and directed GRNs determined by the ChIP data in *C.albicans*. Each distribution contains 30 GRN samples. The fully trained GRNs were solely inferred by the transcriptional profiles while the directed GRNs also had been constrained by the ChIP data. Performing equally well on reproducing the transcriptional profiles, the direct GRNs showed a significant increase compared to the fully trained GRNs.

Bibliography

[1] Irene Cantone, Lucia Marucci, Francesco Iorio, Maria Aurelia Ricci, Vincenzo Belcastro, Mukesh Bansal, Stefania Santini, Mario Di Bernardo, Diego Di Bernardo, and Maria Pia Cosma. A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches. *Cell*, 137(1):172–181, 2009.

[2] Aaron D Hernday, Matthew B Lohse, Polly M Fordyce, Clarissa J Nobile, Joseph L DeRisi, and Alexander D Johnson. Structure of the transcriptional network controlling white-opaque switching in c andida albicans. *Molecular microbiology*, 90(1):22–35, 2013.

[3] Matthew B Lohse, Iuliana V Ene, Veronica B Craik, Aaron D Hernday, Eugenio Mancera, Joachim Morschhäuser, Richard J Bennett, and Alexander D Johnson. Systematic genetic screen for transcriptional regulators of the candida albicans white-

opaque switch. *Genetics*, 203(4):1679–1692, 2016.

[4] Namkha Nguyen, Morgan MF Quail, and Aaron D Hernday. An efficient, rapid, and recyclable system for crispr-mediated genome editing in candida albicans. *MSphere*, 2(2):e00149–17, 2017.

[5] Pamela Moll, Michael Ante, Alexander Seitz, and Torsten Reda. Quantseq 3' mrna sequencing for rna quantification. *Nature methods*, 11(12):i–iii, 2014.

[6] Alexander Dobin and Thomas R Gingeras. Mapping rna-seq reads with star. *Current protocols in bioinformatics*, 51(1):11–14, 2015.

[7] Sascha H Duttke, Max W Chang, Sven Heinz, and Christopher Benner. Identification and dynamic quantification of regulatory elements using total rna. *Genome research*, 29(11):1836–1846, 2019.

[8] Pedro T Monteiro, Jorge Oliveira, Pedro Pais, Miguel Antunes, Margarida Palma, Mafalda Cavalheiro, Mónica Galocha, Cláudia P Godinho, Luís C Martins, Nuno Bourbon, et al. Yeastract+: a portal for cross-species comparative genomics of transcription regulation in yeasts. *Nucleic acids research*, 48(D1):D642–D649, 2020.

[9] Joel F Moxley, Michael C Jewett, Maciek R Antoniewicz, Silas G Villas-Boas, Hal Alper, Robert T Wheeler, Lily Tong, Alan G Hinnebusch, Trey Ideker, Jens Nielsen, et al. Linking high-resolution metabolic flux phenotypes and transcriptional regulation in yeast modulated by the global regulator gcn4p. *Proceedings of the National Academy of Sciences*, 106(16):6477–6482, 2009.

[10] Emily J Parnell, Yaxin Yu, Rafael Lucena, Youngdae Yoon, Lu Bai, Douglas R Kellogg, and David J Stillman. The rts1 regulatory subunit of pp2a phosphatase controls expression of the ho endonuclease via localization of the ace2 transcription factor. *Journal of Biological Chemistry*, 289(51):35431–35437, 2014.

[11] Michael J Carrozza, Laurence Florens, Selene K Swanson, Wei-Jong Shia, Scott Anderson, John Yates, Michael P Washburn, and Jerry L Workman. Stable incorporation of sequence specific repressors ash1 and ume6 into the rpd3l complex. *Biochimica et Biophysica Acta (BBA)-Gene Structure and Expression*, 1731(2):77–87, 2005.

[12] Helen J McBride, Yaxin Yu, and David J Stillman. Distinct regions of the swi5 and ace2 transcription factors are required for specific gene activation. *Journal of Biological Chemistry*, 274(30):21029–21036, 1999.

[13] Robert M Yarrington, Jenna M Goodrum, and David J Stillman. Nucleosomes are essential for proper regulation of a multigated promoter in saccharomyces cerevisiae. *Genetics*, 202(2):551–563, 2016.

[14] Hilary Phenix, Katy Morin, Cory Batenchuk, Jacob Parker, Vida Abedi, Liu Yang, Lioudmila Tepliakova, Theodore J Perkins, and Mads Kærn. Quantitative epistasis analysis and pathway inference from genetic interaction data. *PLoS computational biology*, 7(5):e1002048, 2011.

[15] Raphael Dutoit, Evelyne Dubois, and Eric Jacobs. Selection systems based on dominant-negative transcription factors for precise genetic engineering. *Nucleic acids research*, 38(19):e183–e183, 2010.

[16] Jüri Reimand, Juan M Vaquerizas, Annabel E Todd, Jaak Vilo, and Nicholas M Luscombe. Comprehensive reanalysis of transcription factor knockout expression data in saccharomyces cerevisiae reveals many new targets. *Nucleic acids research*, 38(14):4768–4777, 2010.

[17] Laetitia Cormier, Regine Barbey, and Laurent Kuras. Transcriptional plasticity through differential assembly of a multiprotein activation complex. *Nucleic acids research*, 38(15):4998–5014, 2010.

[18] Matthew B Lohse and Alexander D Johnson. Identification and characterization of wor4, a new transcriptional regulator of white-opaque switching. *G3: Genes, Genomes, Genetics*, 6(3):721–729, 2016.

[19] Aaron D Hernday, Matthew B Lohse, Clarissa J Nobile, Liron Noiman, Clement N Laksana, and Alexander D Johnson. Ssn6 defines a new level of regulation of white-opaque switching in candida albicans and is required for the stochasticity of the switch. *MBio*, 7(1):e01565–15, 2016.

CHAPTER V
DISCUSSION AND CONCLUSION

5.1 Conclusion

In this work, we extended the attractor-matching strategy from a Boolean model to an ODE-based model by incorporating transcriptional kinetic parameters. We consider transcriptional profiles of stable cell types as fixed-point attractors ([1]) in the mRNA state space, and search for the GRN architecture that produces these attractors. We found in the *in silico* simulation that GRN architectures are significantly correlated with the attractors they produce. This correlation supports the logic of applying the attractor-matching approach to GRN inference. The ability of our approach to infer "unknown" GRNs has been validated using both simulated datasets derived from "known" *in silico* GRNs and *in vivo* test datasets from an engineered GRN in *S. cerevisiae*. Our approach outperformed six other leading GRN inference methods when applied to the *in silico* attractors generated by SynTReN ODEs. In the *in vivo* test, our approach not only successfully identified five of the six intended transcriptional edges, but also revealed some unintended edges that might account for the inconsistency between the designed GRN and experimentally derived transcriptional profiles. In addition to inferring GRN architecture based on transcriptional profiles, our approach can also predict the effects of genetic perturbation on the inferred GRN. As a proof of principle, we used the inferred GRNs generated during our *in silico* model testing to then predict the unknown attractors that would be produced upon genetic perturbation of the original reference GRN (i.e., by deletion of each TF). The inferred *in silico* GRNs successfully predicted 71.1% of the attractors produced by the reference GRNs using the identical knockout strains (Table 3.5), indicating that our approach can effectively capture GRN behavior based on transcriptional profiles. This result further suggests that our approach

can be used to generate testable predictions on the behavior of *in vivo* GRNs. Specifically, we envision the application of this approach to a hybrid computational and *in vivo* experimental process whereby GRNs are inferred based on *in vivo* transcriptional profiles, the inferred GRNs are perturbed *in silico* to generate "mutant" transcriptional profiles, and the accuracy of inferred GRNs are ultimately assessed by comparing predicted versus observed transcriptional profiles generated using *in silico* versus *in vivo* mutant strains. The accuracy of the inferred GRN could thus be supported if the predicted and experimentally measured transcriptional profiles converge. If not, the *in vivo* mutant strain and the resulting experimentally derived attractors could reveal a new pattern of GRN dynamics that had not been covered by the initial input attractors, and would thus complement the original wildtype attractors to further refine the inferred GRN. In this manner, it should be possible to iteratively refine predicted GRNs until they approximate the *in vivo* results.

As a proof of principle, we applied this iterative computational and experimental strategy to infer the GRN governing the white-opaque switch in *C. albicans*. We first used a dropout strategy to infer GRNs based on a subset of available data and tested the ability of the inferred GRNs to predict the transcriptional profiles that were omitted from the training data. This approach led to an overall success rate of 66.7%, which approaches the 71.1% success rate observed in our *in silico* testing. Next, we demonstrated that a "fully trained" GRN inferred from all fifteen of the wildtype and single gene deletion strain profiles was successful in predicting the transcriptional profiles of two distinct "unknown" double TF knockout strains that were omitted from our training data. This result demonstrates that the inference of a GRN using a set of known attractors can bring insight into attractors that exist biologically but have not yet been measured in the lab. Although the inferred white-opaque GRNs accurately predicted most if not all of the dropped-out transcriptional profiles, they did not fully converge upon the TF localization patterns that we have observed in *in vivo* genome-wide TF localization experiments. To further constrain the white-opaque GRN, we inferred the "directed" GRNs with all seventeen available experimentally measured transcriptional profiles and included ChIP data that biases the GRN architecture towards the TF localization pattern observed *in vivo*. This consensus directed GRN accurately reproduced 76% of the RNAseq-derived transcriptional profiles and converged upon 99% of the ChIP-derived TF binding interactions. This directed GRN model can be iteratively tested and refined in the future and to provide further insight into the transcriptional regulatory dynamics of this highly intertwined and complex GRN that controls cellular differentiation in *C. albicans*.

5.2 Limitations and challenges

There are potential pitfalls that can impact our approach. First, regulatory elements other than TFs, such as non-coding RNA molecules, post-translational modifications, and chromatin modifiers/remodelers, can also influence the behavior of a GRN of interest. Their regulatory effects can lead to false compensatory TF regulations and make the inferred network converge less often. Second, as observed in our *C. albicans* GRN modeling, noise in the experimental data can lead to a "fuzzy" target for prediction and compromise the ability of the approach to fit the transcriptional profiles into a GRN. Furthermore, while RNAseq data derived from a particular genotypic/phenotypic state is assumed to represent a fixed-point attractor, this is not necessarily the case. Multiple stable states or oscillatory transcriptional outputs could exist within a population of cells that appear to be phenotypically homogeneous, thus bulk RNA sequencing could average out single-cell heterogeneity and underlying GRN dynamics. These limitations could lead to inferred GRNs that simulate biased or non-existent targets. Third, our approach assumes that the highest expression levels for each gene have been observed in the input transcriptional profiles and utilizes them to estimate the unknown parameters. Potential bias in parameter estimation can occur if this assumption is not satisfied. Moreover, our approach has simplified the way that multiple activators or inhibitors regulate a target gene, either independently as monomers or cooperatively as a polymer, but *in vivo* TFs could have more complex and sophisticated forms of incorporation than modeled in our approach. These and likely other confounding factors have the potential to adversely impact the process of GRN inference and can cause reduced accuracy in predicting unknown transcriptional profiles.

The most significant challenge in GRN inference is perhaps the inherent functional redundancy and plasticity of real-world GRNs. This was apparent in our *in silico* testing where we observed that GRNs differing in as many as ten regulatory interactions can produce qualitatively similar transcriptional profiles (Fig. 3.4). Similarly, we observed that most of the attractors produced by the *C. albicans* white-opaque GRN could be reproduced, and "unknown" attractors predicted, even when the inferred GRN does not closely match the experimentally determined GRN architecture (Fig. 4.2). These results are consistent with the idea that GRN structures can evolve while maintaining the same overall output, which is also supported by experimental evidence ([2]). For example, Tsong et al. ([3]) identified a set of sexual differentiation genes that are negatively regulated in *S. cerevisiae*, but are believed to have been positively regulated in an ancestral fungal species. In this example, the overall output of the transcriptional circuit remains the same, despite significant

changes in GRN architecture. Our work provides a mathematical foundation for the idea that GRN architecture has plasticity and evolves ([4, 5, 6]) under selective pressure ([7]). Thus, experiments performed under a specific set of experimental conditions may fail to reveal some of the evolutionary pressures that have constrained the behavior of real-world GRNs under distinct environmental conditions. While the impact of these unobserved evolutionary pressures on GRN architecture and logic could be revealed by extensive measurements of GRN output in an array of different environmental conditions, we hypothesize that the iterative model refinement strategy that we propose here may represent an efficient alternative strategy.

5.3 Future work

In future iterations of our GRN inference approach, we can incorporate other types of interactions between TFs that are not independent as assumed by default. For instance, to consider the fact that Gal80 can only perform gene regulation by binding to Gal4, we can add the following rule to the algorithm: if a target gene is regulated by Gal80 but not by Gal4, the regulation of Gal80 on this gene will be voided. Interactions between metabolites and TFs, such as IPTG deactivating the lac repressor, can also be incorporated into the approach by adding similar rules. In this manner, our approach can flexibly integrate more detailed biological information beyond sequencing data and better simulate complex biological systems.

Another future attempt would be to apply our approach to the full GRN governing the white-opaque switch in *C. albicans*. We aim to select all the genes that impact the frequency of the white-opaque switch and create a set of mutant strains accordingly, such as gene knock-out mutants and transcription factor binding site knock-out mutants. We will apply our approach and the iterative refinement strategy to infer a GRN that matches the behavior of these mutant strains and agrees with the ChIP and csRNA sequencing data. This inferred GRN can shed light on the full picture of the regulatory interactions amongst the white-opaque specific transcriptional regulators.

Bibliography

[1] Stuart A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3):437–467, 1969. ISBN: 0022-5193 Publisher: Academic Press.

[2] Chiraj K Dalal and Alexander D Johnson. How transcription circuits explore alternative architectures while maintaining overall circuit output. *Genes & Development*, 31(14):1397–1405, 2017.

[3] Annie E Tsong, Brian B Tuch, Hao Li, and Alexander D Johnson. Evolution of alternative transcriptional circuits with identical logic. *Nature*, 443(7110):415–420, 2006.

[4] Isabel Nocedal, Eugenio Mancera, and Alexander D Johnson. Gene regulatory network plasticity preyears a switch in function of a conserved transcription regulator. *Elife*, 6:e23250, 2017.

[5] Christopher R Baker, Lauren N Booth, Trevor R Sorrells, and Alexander D Johnson. Protein modularity, cooperative binding, and hybrid regulatory states underlie transcriptional network diversification. *Cell*, 151(1):80–95, 2012.

[6] Douglas H. Erwin. Chapter thirteen - evolutionary dynamics of gene regulation. In Isabelle S. Peter, editor, *Gene Regulatory Networks*, volume 139 of *Current Topics in Developmental Biology*, pages 407–431. Academic Press, 2020.

[7] Trevor R Sorrells and Alexander D Johnson. Making sense of transcription networks. *Cell*, 161(4):714–723, 2015.

Appendix

I Numerical solution and dynamical errors

In order to articulately describe the notions of numerical solution and dynamical error, we take a single pendulum system as an example. As shown in Fig. 2.4 a, a pendulum is a massive ball connected to a pivot by a massless rod. Descriptions and units for the symbols used in the pendulum model are listed in Table S1. The state variables of the pendulum model would be the amplitude θ, which indicates the current location of the ball, and the angular velocity ω, which tells how fast the ball is moving. We know that the pendulum is pulled by gravity and the rod at the same time. When the pendulum moves, it will also receive a resistance if friction exists ($\mu \neq 0$). With this information, we can write down the equation below.

$$mL^2\frac{d^2\theta}{dt^2} + mgL\sin\theta + mL^2\mu\frac{d\theta}{dt} = 0, \tag{S1}$$

where the first term is the total torque of the pendulum, the second term is the torque created by gravity, and the last term is the torque created by resistance.

Differential equations can be solved numerically or analytically. Numerical methods use finite numbers to compute the integrals defined by the ODEs step by step. While analytical methods, also known as symbolic computation, use symbols, such as x and y, to compute the solutions. Therefore, an analytical solution is perfect because it has absolutely no error. However, it has been proved that Eq. S1 cannot be solved analytically (i.e. in closed-form expression) when $\mu \neq 0$ ([1]). Specifically, if a person wants to write down the analytical solution of Eq. S1, if it exists, with normal operations and functions, such as plus, minus, multiplication, exponent, logarithm, and trigonometric functions, he must need infinite amount of ink. In fact, analytical solution is not practical for most complex

non-linear differential equations, and numerical solution is the only option to solve them. For example, given an initial state of the pendulum (θ_0, ω_0) at time t_0, we can apply the Eq. S2 to calculate the value of ω_t the at time $t = t_0 + \delta t$, where δt cannot be infinitesimal since computers do not have infinite memories to store or calculate it. Knowing the values of θ_0, ω_t, and δt, we can use Eq. S2 to calculate the value of θ_t.

$$\theta_t = \theta_0 + \omega_t \cdot \delta t. \tag{S2}$$

By calculating the values of ω_t and θ_t step by step, we can obtain a numerical solution of Eq. S1, or a trajectory as shown in Fig. S1b, with respect to the initial condition (θ_0, ω_0), and this method (Eq. S2) is called forward Euler. However, error is inevitable when using numerical methods to solve ODEs. As shown in Fig. S1a, since we assumed that the values of the derivative, which is ω_t in this case, do not change over the time interval δt, integrating θ_t using Eq. S2 will lead to a deviation from the correct trajectory, and this is called the dynamical error. What's worse, the dynamical error can, and will, snowball as the integration goes on. Generally speaking, as the value of δt and the times of integration increase, the dynamical error will become larger, and it can create or break the attractors of a dynamic system. As shown in Fig. S1b, by analytically solving the Eq. S1 when $\mu = 0$, we know that there is a periodic attractor, also known as a limit cycle, in the phase space. However, when we solve the same equation using numerical methods, in this case forward Euler (Eq. S2), the periodic attractor disappears. Therefore, it is important to make sure that the dynamical error caused by the numerical method does not significantly affect our result.

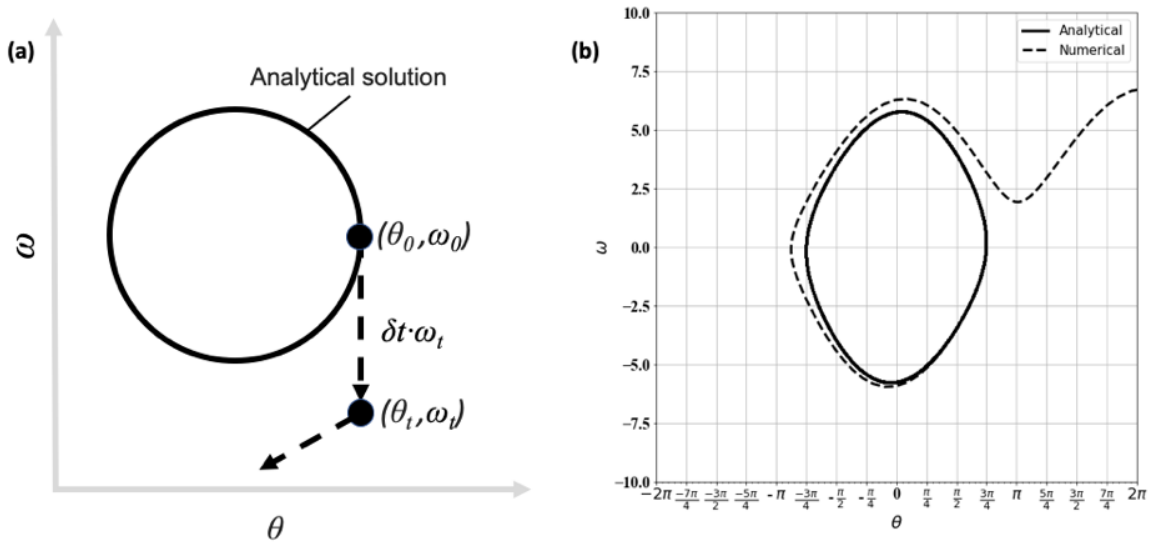


Figure S1: (a) Schematic of how dynamical error leads to a deviation from the true trajectory. The cycle represents the true trajectory solved by analytical method. The dashed arrows indicate the integration steps in numerical methods. (b) An example in the case of a single pendulum model, whose parameters are $g = 9.8$ m/s^2, $\mu = 0$ kg/s, $m = 1$ kg and $L = 1$ m. The initial state is $(\theta_0 = 3\pi/4, \omega_0 = 0)$. The solid line is calculated by analytical method while the dashed line is by numerical method (forward Euler). The dynamical error caused by numerical methods can lead to divergence and break the periodic attractor (i.e. the limit cycle).

Table S1: Parameter table for the pendulum dynamic system

Symbol	Description	Unit
m	mass of the ball	kilogram
L	length of the rod	meter
θ	the angle between the pendulum and the vertical line	radian
ω	the angular velocity of the ball	radian/second
μ	the damping coefficient	kilogram/second
g	the gravitational acceleration	meter/second^2

Table S2: Probabilities of cumulative attractor distance by the null model

	Attractor distances				
Number of genes	≤ 0.1	≤ 0.15	≤ 0.2	≤ 0.3	≤ 0.4
5 genes	0.61%	3.81%	12.50%	45.10%	73.39%
6 genes	0.29 %	2.58 %	10.53%	44.77%	74.11%
7 genes	0.14 %	1.79 %	9.00%	44.55 %	74.68%
8 genes	0.07%	1.25%	7.77%	44.36%	75.08%
9 genes	0.03%	0.89%	6.75%	44.23%	75.42%

Table S2 shows the probabilities of cumulative attractor distances produced by a null model. For each gene, the null model randomly picks a value in a continuous uniform distribution $\mathbf{U}([R]_{i,min}, [R]_{i,max})$, where $[R]_{i,min}$ and $[R]_{i,max}$ are the minimal and maximal expression levels of the i^{th} gene.

Table S3: *C. albicans* strains used in this study

Description	AHY	TF	Genotype	Reference
*a/*Δ *wildtype*	304	WT	*a/*Δ*MTLalpha::ARG4 C.m.LEU2/*Δ*leu2 C.d.HIS1/*Δ*his1 URA3/ura3D::imm*^{434} *IRO1/iro1*Δ*::imm*^{434} *arg4::hisG/arg4::hisG*	[2]
Δ/Δ*wor1*	856	Wor1	*a/*Δ*MTLalpha::ARG4 C.m.LEU2/*Δ*leu2 C.d.HIS1/*Δ*his1 URA3/ura3D::imm*^{434} *IRO1/iro1*Δ*::imm*^{434} *arg4::hisG/arg4::hisG* Δ*orf19.4884(wor1)::C.a.HIS1/*Δ*orf19.4884(wor1)::C.a.LEU2*	[2]
Δ/Δ*wor2*	736	Wor2	*a/*Δ*MTLalpha::ARG4 C.m.LEU2/*Δ*leu2 C.d.HIS1/*Δ*his1 URA3/ura3D::imm*^{434} *IRO1/iro1*Δ*::imm*^{434} *arg4::hisG/arg4::hisG* Δ*orf19.5992(wor2)::C.a.HIS1/*Δ*orf19.5992(wor2)::C.a.LEU2*	[2]

Δ/Δ*wor3*	850	Wor3	*a/ΔMTLalpha::ARG4 C.m.LEU2/Δleu2 C.d.HIS1/Δhis1 URA3/ura3D::imm^{434} IRO1/iro1Δ::imm^{434} arg4::hisG/arg4::hisG Δorf19.467(wor3)::C.a.HIS1/Δorf19.467(wor3)::C.a.LEU2*	[2]
Δ/Δ*wor4*	861	Wor4	*a/ΔMTLalpha::ARG4 C.m.LEU2/Δleu2 C.d.HIS1/Δhis1 URA3/ura3D::imm^{434} IRO1/iro1Δ::imm^{434} arg4::hisG/arg4::hisG Δorf19.6713(wor4)::C.a.HIS1/Δorf19.6713(wor4)::C.a.LEU2*	[2]
Δ/Δ*efg1*	836	Efg1	*a/ΔMTLalpha::ARG4 C.m.LEU2/Δleu2 C.d.HIS1/Δhis1 URA3/ura3D::imm^{434} IRO1/iro1Δ::imm^{434} arg4::hisG/arg4::hisG Δorf19.610(efg1)::C.a.HIS1/Δorf19.610(efg1)::C.a.LEU2*	[2]
Δ/Δ*ahr1*	812	Ahr1	*a/ΔMTLalpha::ARG4 C.m.LEU2/Δleu2 C.d.HIS1/Δhis1 URA3/ura3D::imm^{434} IRO1/iro1Δ::imm^{434} arg4::hisG/arg4::hisG Δorf19.7381(ahr1)::C.a.HIS1/Δorf19.7381(ahr1)::C.a.LEU2*	[2]
Δ/Δ*czf1*	784	Czf1	*a/ΔMTLalpha::ARG4 C.m.LEU2/Δleu2 C.d.HIS1/Δhis1 URA3/ura3D::imm^{434} IRO1/iro1Δ::imm^{434} arg4::hisG/arg4::hisG Δorf19.3127(czf1)::C.a.HIS1/Δorf19.3127(czf1)::C.a.LEU2*	[2]
Δ/Δ*ssn6*	801	Ssn6	*a/ΔMTLalpha::ARG4 C.m.LEU2/Δleu2 C.d.HIS1/Δhis1 URA3/ura3D::imm^{434} IRO1/iro1Δ::imm^{434} arg4::hisG/arg4::hisG Δorf19.6798(ssn6)::C.a.HIS1/Δorf19.6798(ssn6)::C.a.LEU2*	[2]
Δ/Δ*rbf1*	793	Rbf1	*a/ΔMTLalpha::ARG4 C.m.LEU2/Δleu2 C.d.HIS1/Δhis1 URA3/ura3D::imm^{434} IRO1/iro1Δ::imm^{434} arg4::hisG/arg4::hisG Δorf19.5558(rbf1)::C.a.HIS1/Δorf19.5558(rbf1)::C.a.LEU2*	[2]

Δ/Δ*wor1*	1355	Wor1	*a/Δalpha C.m.LEU2/Δleu2 C.d.HIS1/his1Δ*
Δ/Δ*ssn6*		Ssn6	*URA3/ura3D::imm*^{434} *IRO1/iro1Δ::imm*^{434} *arg4::hisG/arg4::hisG ΔMTLalpha::ARG4 Δorf19.6798(ssn6)::C.a.HIS1/Δorf19.6798(ssn6)::C.a.LEU2 Δwor1/Δwor1*
Δ/Δ*wor1*	1354	Wor1	*a/Δalpha C.m.LEU2/leu2Δ C.d.HIS1/his1Δ*
Δ/Δ*rbf1*		Rbf1	*URA3/ura3D::imm*^{434} *IRO1/iro1Δ::imm*^{434} *arg4::hisG/arg4::hisG ΔMTLalpha::ARG4 Δorf19.5558(rbf1)::C.a.HIS1/Δorf19.5558(rbf1)::C.a.LEU2 Δwor1/Δwor1*

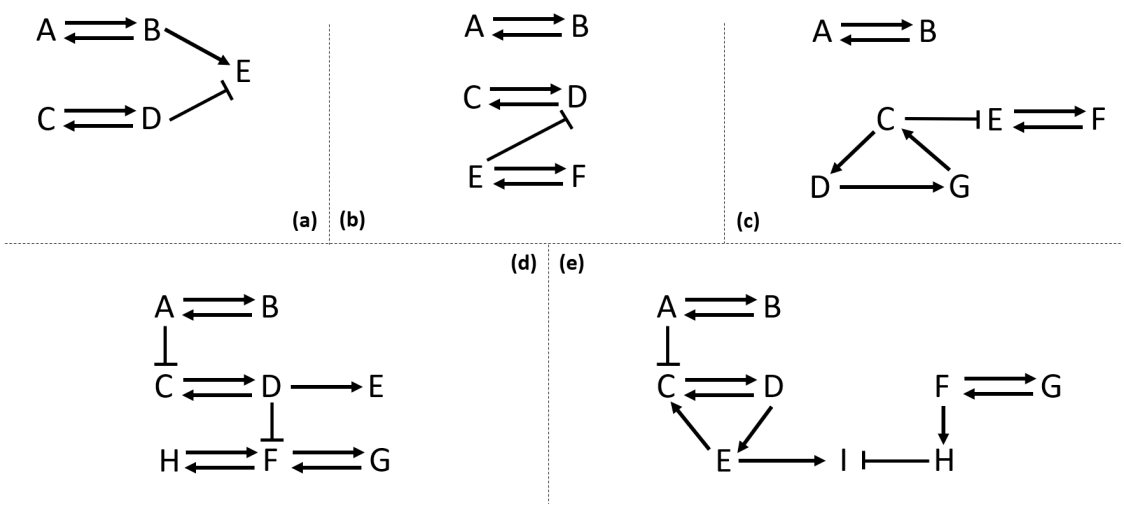


Figure S2: Five GRN architectures were arbitrarily generated as references in the *in silico* test. They have 5-9 (a-e) genes and no self-regulatory edges. The pointed and (or blunt) arrows represent activating (and repressing) regulatory interactions, respectively.

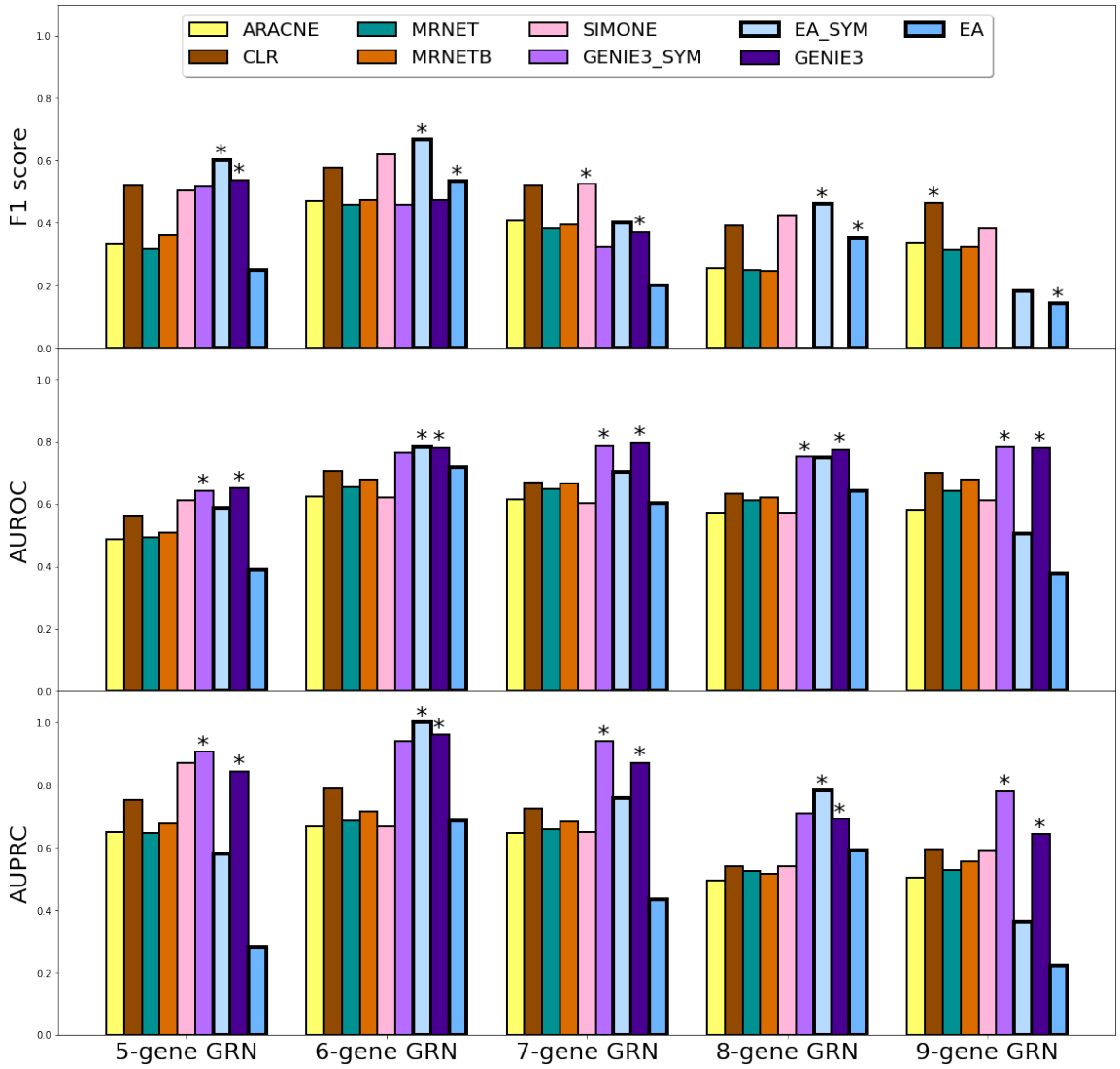


Figure S3: The non-autoregulation *in silico* test comparison results in F1 score (upper panel), AUROC (middle panel), and AUPRC (bottom panel). The F1 scores are calculated using a threshold cutoff of 0.5 for all models. The best performance is marked by a star for symmetric and asymmetric methods.

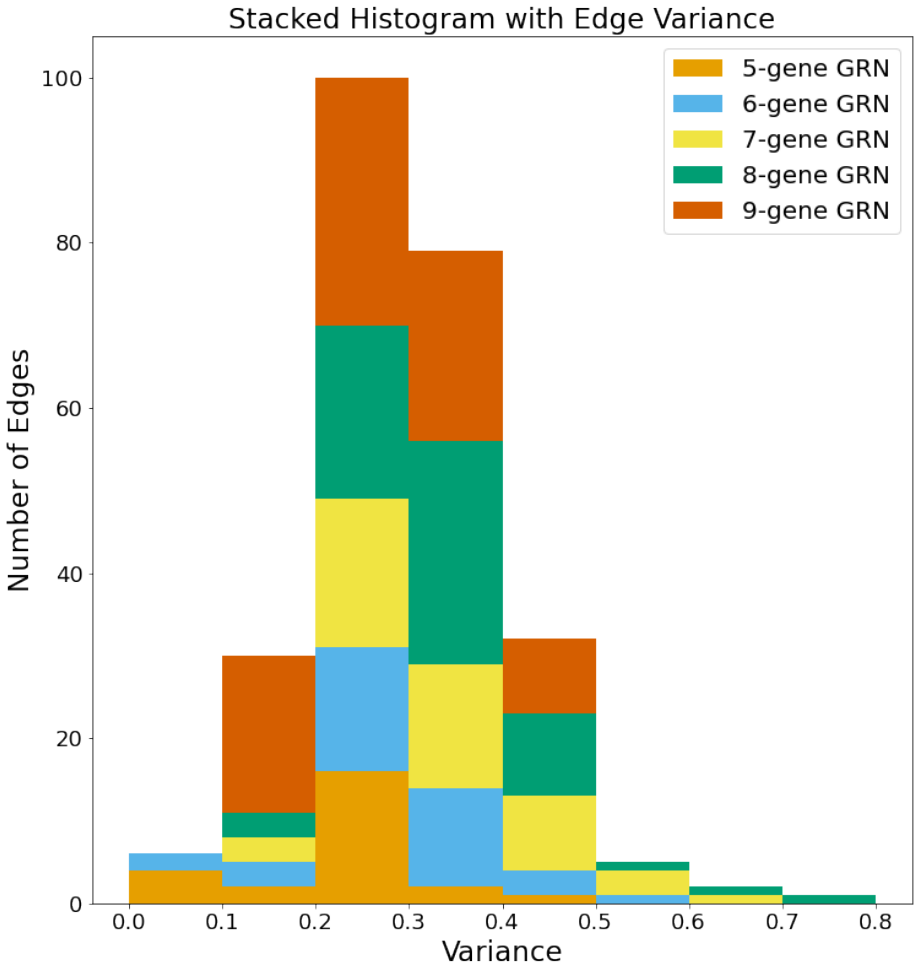


Figure S4: A stacked histogram displays the distribution of edge variances across 30 independent inference runs, showing the number of edges for each variance category.

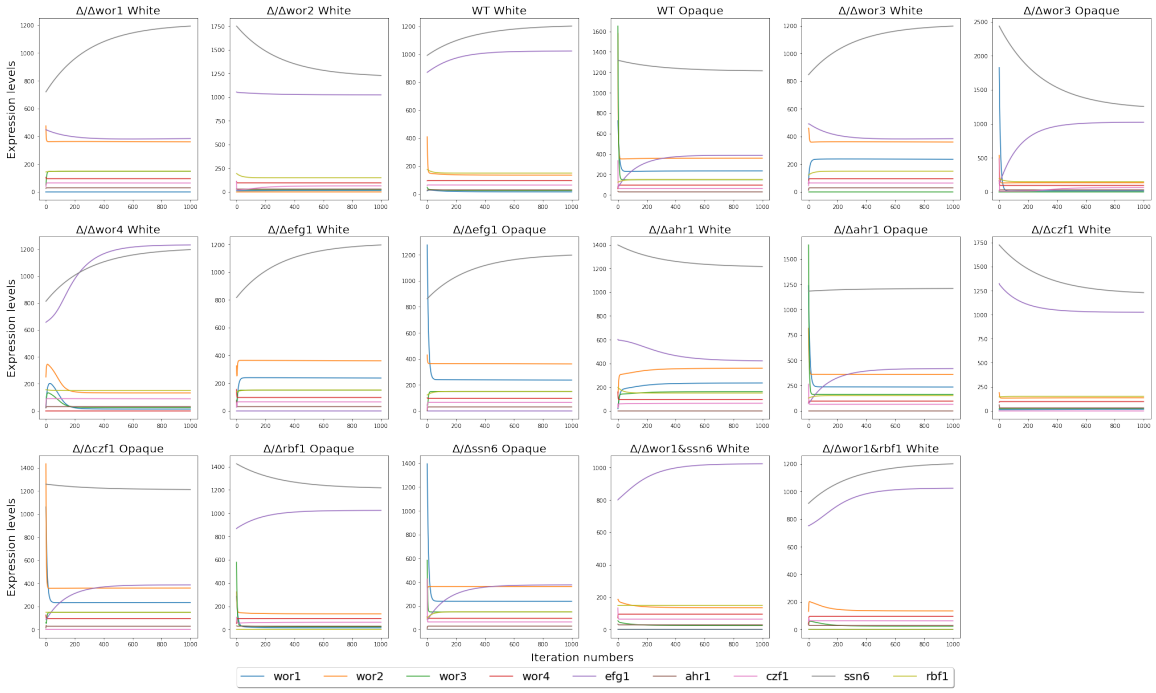


Figure S5: Prediction of drop-out transcriptional profiles in *C. albicans*. A dropout strategy was utilized to infer GRNs based on a subset of available data and assessed the predictive capability of the inferred GRNs for transcriptional profiles that were deliberately excluded from the training dataset. The initial states were configured to correspond to the omitted transcriptional profiles.

Bibliography

[1] Josh Bevivino. The path from the simple pendulum to chaos. *Dynamics at the Horsetooth*, 1(1):1–24, 2009.

[2] Matthew B Lohse, Iuliana V Ene, Veronica B Craik, Aaron D Hernday, Eugenio Mancera, Joachim Morschhäuser, Richard J Bennett, and Alexander D Johnson. Systematic genetic screen for transcriptional regulators of the candida albicans white-opaque switch. *Genetics*, 203(4):1679–1692, 2016.