Cell-type Specific Gene Regulatory Network Inference In Mouse Embryonic Development

EDUARD MA[1,*]

[1] Delft Bioinformatics Lab, Delft University of Technology, Delft 2628XE, The Netherlands
[*] e.e.w.ma@student.tudelft.nl

Compiled March 11, 2024

Exploring gene-gene interactions during embryonic development has been pivotal across various microbiological sub-fields, from gene functional annotation to understanding pathogenesis. Although numerous methods exist for inferring these interactions from time-series gene-expression data, they often lack a coherent biophysical foundation. In this study, we implement a gene-gene interaction inference pipeline grounded in statistical thermodynamics principles. Evaluating its efficacy on synthetic data and mouse single-cell RNA sequencing embryonic developmental data, we attempted to unveil, for the first time, the gene-gene interactions at play in murine embryos through the lens of statistical thermodynamics.

1. INTRODUCTION

Since the introduction of next-generation sequencing (NGS), it has become possible to sequence thousands of genes simultaneously—the so-called high-throughput aspect of NGS. This opens up the possibility of analyzing how different genes, through activation or inhibition, interact with each other. Elucidating gene-gene interactions enhances our understanding of life, spanning from low-level cellular processes to the top-level morphology of an organism, and is essential in all subfields of biology, most prominently in biomedicine [1, 2].

Many methods have been developed to infer gene-gene interactions from genomic data [3]. Of particular interest is time-series data, where the gene expression of different genes is measured discretely at various time points. Intuitively, for a two-gene system, if the expression of gene A increases over time while the expression of gene B decreases, we can establish that gene A inhibits the expression of gene B. Similar reasoning is employed to establish the activation of genes. For a larger number of genes, inferring the exact gene-gene interactions becomes more complex. Extending this logic to Next-Generation Sequencing (NGS) data, inferring the interactions from the measurement of thousands of gene-expression profiles per sample is nearly impossible by hand [4]. Automation of this process, therefore, becomes essential as the volume of data increases to discover more complex gene expression patterns.

To address these challenges, Zamanighomi *et al.* have developed a gene-gene interaction network inference pipeline based on the biophysics of dogmatic gene expression [5]. Their method enables the rapid processing of high-volume time-series datasets, as their inference pipeline comes down to solving convex optimization problems through sparse linear regression. Interestingly, their model was designed for gene perturbation data, where gene expression disruption needs to occur to induce unnatural gene expression dynamics, which is subsequently used to infer gene-gene interactions. Acquiring this data in the wet lab is an exceptionally laborious task, and for many organisms, such data is unavailable. However, what is available is time-series single-cell RNA sequencing (scRNA-seq) data of embryonic development [6, 7]. During embryonic development, cells differentiate into specialized entities, each with distinctive morphology and corresponding gene expression patterns [8]. To some extent, this dynamic process mimics gene expression disruption, as external signals often induce significant differences in gene expression to drive cell differentiation. It is, therefore, interesting to see if gene-gene interactions can be inferred from embryonic data to shed light on how developmental genes interact during processes like gastrulation and organogenesis.

In this honors project, we implemented the gene-gene interaction inference pipeline devised by *Zamanighomi et al.*, adapting it to analyze gene-gene interactions in murine embryonic developmental single-cell RNA sequencing (sc-RNA-seq) data [5]. The pipeline relies on a simplified biophysics model inspired by *Bintu et al.* [9]. The pipeline's performance is assessed using both synthetic and authentic developmental embryonic data. To the best of the author's knowledge, this biophysics model has not been previously applied to embryonic data. Furthermore, this report serves as both a test of the inference pipeline's effectiveness with non-perturbed time-series gene expression data and as an exploration of the gene-gene interactions discovered during the embryogenesis of mice.

2. MODEL

In the simplified version of the model by [5], we simulate gene expression levels following the central dogma of molecular biology. This process involves the transcription of genes into mRNA, and the subsequent translation of mRNA into proteins [10]. These proteins may then function as regulatory agents, serving as either activators or repressors for other genes.

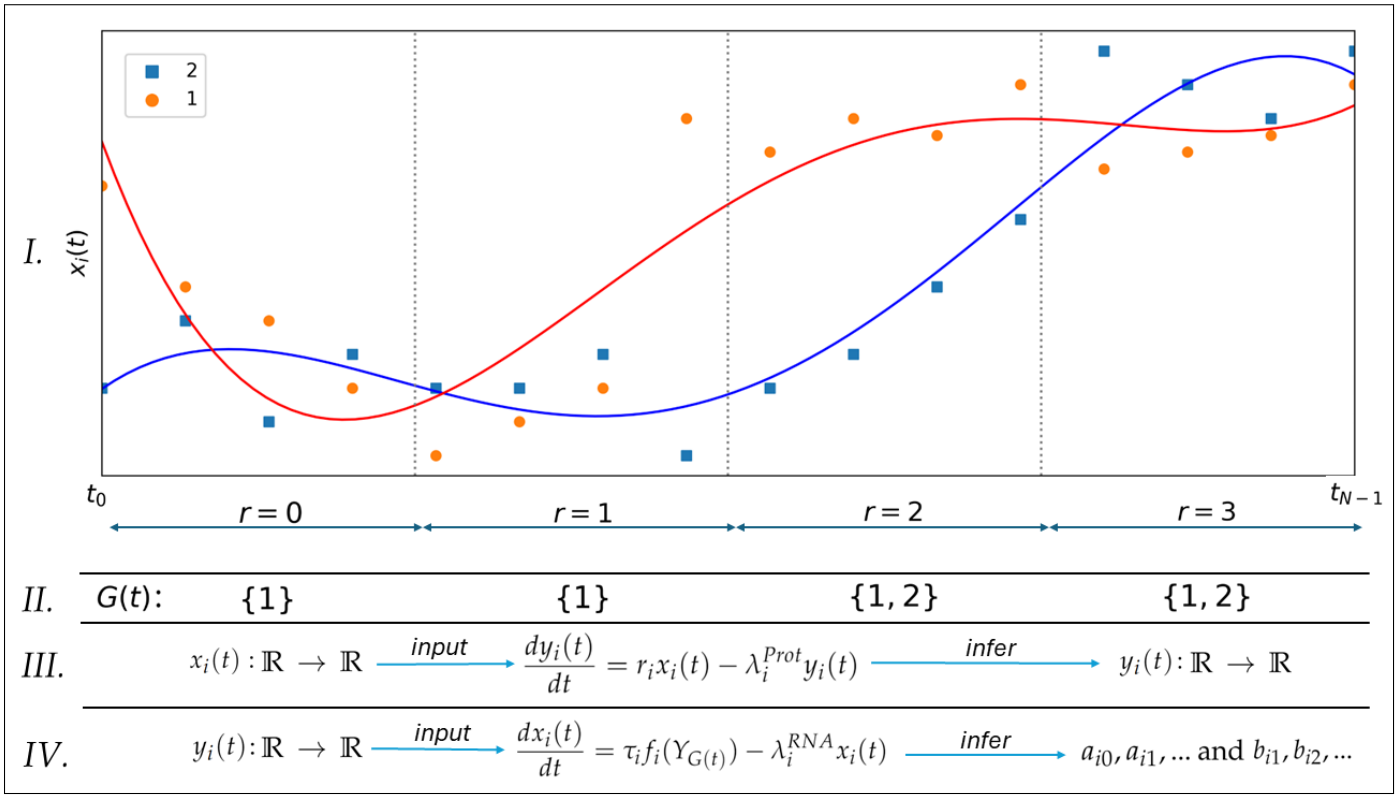


Fig. 1. inference pipeline flow in 4 steps. We first interpolate the discrete gene-expression measurement points to remove noise. Next, we consider which genes deviate significantly from stead-state gene-expression and add them to $G(t)$. These genes are considered as possible regulators on all other genes in the system. We then solve two convex optimization problems to infer the protein expression function $y_i(t)$. Lastly, the second convex optimization problem is solved in which we estimate the contribution of every possible gene regulator to the gene expression of every gene in the system. These coefficients represent the final inferred gene-network. $R = 4$ and $T = 0.3$

A. Data

The input to the model consists of time-course RNA gene expression data, denoted as matrix $\mathbf{X} \in \mathbb{R}^{M\times N}$, structured as follows:

$$\mathbf{X} = \begin{bmatrix} x_0(t_0) & x_o(t_1) & \cdots & x_0(t_{N-1}) \\ x_1(t_0) & x_1(t_1) & \cdots & x_1(t_{N-1}) \\ \vdots & \vdots & \ddots & \vdots \\ x_{M-1}(t_0) & x_{M-1}(t_1) & \cdots & x_{M-1}(t_{N-1}) \end{bmatrix}$$

Here, $x_i(t_j)$ represents the RNA expression level of gene i at time point t_j, with $i \in 0, 1, ..., M-1$ and $j \in 0, 1, ..., N-1$. The model then derives significant gene-gene interactions from M genes observed over N time points. It is essential that M > N for subsequent regression analysis, which is a valid requirement given the abundance of gene expression data in the order of hundreds to thousands [11].

Additionally, protein expression $\mathbf{Y} \in \mathbb{R}^{M\times N}$ is subsequently inferred from $\mathbf{X}$ and organized as follows:

$$\mathbf{Y} = \begin{bmatrix} y_0(t_0) & y_o(t_1) & \cdots & y_0(t_{N-1}) \\ y_1(t_0) & y_1(t_1) & \cdots & y_1(t_{N-1}) \\ \vdots & \vdots & \ddots & \vdots \\ y_{M-1}(t_0) & y_{M-1}(t_1) & \cdots & y_{M-1}(t_{N-1}) \end{bmatrix}$$

It's important to note that $\mathbf{Y}$ is not assumed to be an input dataset for the model.

B. Biophysics

Every gene is assumed to follow the simplified central dogma model outlined in [8]. In other words, each gene is characterized by two first-order differential equations that collectively describe the concentrations of $x_i(t)$ and $y_i(t)$ (Equations 1 and 3, respectively). Initially, mRNA is transcribed from DNA, and Equation 1 captures the dynamics of mRNA expression for any given gene over time.

$$\frac{dx_i(t)}{dt} = \tau_i f_i(Y_{G(t)}) - \lambda_i^{RNA} x_i(t) \quad \textbf{(1)}$$

where $(\tau_i f_i(Y_{G(t)}))$ quantifies mRNA production. Here, τ_i represents the gene-specific transcription rate, and $f_i(Y_{G(t)})$ denotes a weighted fraction of transcription factors bound to the genome at time t (Equation 2). Essentially, $f_i(Y_{G(t)})$ characterizes the probability of RNA polymerase binding, and thus transcription occurring, based on the protein expression levels of genes out of steady state up to time point t, denoted as $Y_{G(t)} = y_i(t) | \forall i \in G(t)$.

$$f_i(Y_{G(t)}) = \frac{a_{i0} + \sum_{j=1}^{W(t)} a_{ij} \prod_{k\in S_{ij}(t)} y_k(t)}{1 + \sum_{j=1}^{W(t)} b_{ij} \prod_{k\in S_{ij}(t)} y_k(t)} \quad \textbf{(2)}$$

Here, $W(t)$ signifies the number of first- and second-order regulator complexes influencing the gene expression of a given

gene.

To exemplify, in a two-state system described by imaginary dataset $\mathbf{X}'$ similarly structured in Section 2A, $G(t) = \{1,2\}$, $S_i = (\emptyset, \{1\}, \{2\}, \{1,2\})$, and $W(t) = 4$, representing the set of first- and second-order regulator complexes. The original function is then expressed as:

$$f_i(Y_{G(t)}) = \frac{a_{i0} + a_{i1}y_1(t) + a_{i2}y_2(t) + a_{i3}y_1(t)y_2(t)}{1 + b_{i1}y_1(t) + b_{i2}y_2(t) + b_{i3}y_1(t)y_2(t)}$$

Now, assuming that from $\mathbf{X}'$, we infer gene i to be upregulated by protein 1 and downregulated by protein 2, the system is given as:

$$\frac{dx_i(t)}{dt} = \tau_i \frac{a_{i0} + a_{i1}y_1(t)}{1 + b_{i2}y_2(t)} - \lambda_i^{RNA} x_i(t)$$

The mRNA concentrations in Equation 1 are influenced by the second term, representing basal mRNA degradation in a cell. λ_i denotes the gene-specific basal mRNA degradation constant. When $\frac{dx_i(t)}{dt} > 0$, the mRNA concentration of gene i increases; if $\frac{dx_i(t)}{dt} < 0$, it decreases. At $\frac{dx_i(t)}{dt} = 0$, the cell is in a steady state, and mRNA concentrations remain stable. Similarly, Equation 3 describes the protein level of gene i over time.

$$\frac{dy_i(t)}{dt} = r_i x_i(t) - \lambda_i^{Prot} y_i(t) \tag{3}$$

Here, $r_i x_i(t)$ represents protein production over time, while the right term represents protein degradation. r_i is the mRNA translation rate of gene i, and λ_i^{Prot} is the basal protein degradation rate. The same principles apply to protein concentrations over time as described for mRNA concentrations, where $\tau_i, \lambda_i^{RNA}, r_i, \lambda_i^{Prot} > 0$. microRNA influences as described by *Zamanighomi et al.* are excluded in this project.

3. INFERENCE PIPELINE

The inference pipeline can be broken down into four steps (Figure 1):

I) Estimating RNA-expression levels: we interpolate $\mathbf{X}$ to estimate $x_i(t) : \mathbb{R} \to \mathbb{R}, \forall i \in \{0, 1, ..., M-1\}$.

II) Detecting out-of-steady state genes: through regression analysis, we construct the time-dependent set $G(t)$ to identify out-of-steady state genes. At time t, $G(t)$ represents a set with genes $i \in \{0, 1, ..., M-1\}$ found in the time interval $[0, t)$, assumed to be potential regulators from that point onward.

III) Inferring protein expression: solve Equation 3 for all M genes to infer protein expression $y_i(t)$. In our simplified biophysics model, only the protein products (up to second-order) serve as direct gene regulators. Each gene has its tailored version of Equation 3, representing a linear convex optimization problem.

IV) Estimating coefficients in $f_i(Y_{G(t)})$: solve Equation 1 through another convex optimization problem to estimate the coefficients in $f_i(Y_{G(t)})$. This step uncovers the regulatory behavior that genes in the system exert on each other.

The computationally most expensive part of the pipeline is in step IV where the SLSQP algorithm ($O(W(t)^3)$) is used to solve the final optimization problem [12]. $W(t)$ in the worst case contains all possible regulatory complexes created by M genes up to order two which can be calculated as $W(t) = 2^M - 1 + M$. Hence, part IV has $O(2^{3M})$ time complexity.

The ultimate objective is to estimate the coefficients $a_{i0}, a_{i1}, ...$ and $b_{i1}, b_{i2}, ...$ for all genes $i \in \{0, 1, ..., M-1\}$. The corresponding protein terms linked to these coefficients appearing solely in the denominator, act as repressors, with the degree of repression based on the magnitude of the coefficient. Terms appearing in both the numerator and denominator may function as either activators or repressors, based on their relative magnitudes.

I. mRNA Expression Estimation

The first step involves estimating the gene expression function $x_i(t) : \mathbb{R} \to \mathbb{R}$ for all M genes to filter out noise in the measurement points of $\mathbf{X}$. For each gene, a weighted linear sum of D B-spline basis functions is fitted to the discrete data points.

$$x_i(t) = \sum_{d=1}^{D} \theta_{id}\phi_d(t) = \begin{bmatrix}\phi_1(t) & \cdots & \phi_D(t)\end{bmatrix} \begin{bmatrix}\theta_{i1} \\ \vdots \\ \theta_{iD}\end{bmatrix} = \boldsymbol{\phi}(t)^T \boldsymbol{\theta_i} \tag{4}$$

The i-th B-spline basis function of degree p, $\phi_{i,p}(t)$, is recursively defined as [13]:

$$\phi_{i,0}(t) = \begin{cases} 1 & \text{if } t_i \le t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$\phi_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i}\phi_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}}\phi_{i+1,p-1}(t)$$

To infer the $\boldsymbol{\theta_i}$ weights we solve a regularized L_2 norm minimization problem for every gene i

$$\min_{\boldsymbol{\theta_i}} \left\| \sum_{j=0}^{N-1} (x_i(t_j) - \boldsymbol{\phi}(t)^T \boldsymbol{\theta}_i) \right\|_2 + \gamma_\theta \boldsymbol{\theta}_i^T \mathbf{K} \boldsymbol{\theta}_i, \tag{5}$$

$$K(j,k) = \int_{t_0}^{t_N} \phi_j''(t)\phi_k''(t)dt$$

where $\mathbf{K} \in \mathbb{R}^{D \times D}$ acts as a smoothing factor in the regularization of the fitted B-spline curve. We use the optimized `B-spline` package of `scipy` for calculating the basis function $\phi_d(t)$ non-recursively to enhance computational efficiency [14]. While estimating $x_i(t)$ as a weighted sum of B-spline functions has the advantage of obtaining a noise-free estimate of gene expression, it also provides an immediate expression for the first derivative, $\frac{dx_i(t)}{dt} = x_i'(t)$:

$$\phi_{i,p}'(t) = \frac{p}{t_{i+p} - t_i}\phi_{i,p-1}(t) - \frac{p}{t_{i+p+1} - t_{i+1}}\phi_{i+1,p-1}(t)$$

Thus, the derivative of our gene-expression functions can be expressed as another weighted sum of B-spline basis functions:

$$\frac{dx_i(t)}{dt} = \sum_{d=1}^{D} \theta_{id}\phi_d'(t) = \begin{bmatrix}\phi_1'(t) & \cdots & \phi_D'(t)\end{bmatrix} \begin{bmatrix}\theta_{i1} \\ \vdots \\ \theta_{iD}\end{bmatrix} = \boldsymbol{\phi}(t)'^T \boldsymbol{\theta_i}$$

II. Gene-expression Analysis

We then proceed to construct the time-dependent set $G(t)$ based on the estimated $x_i(t)$ (refer to Algorithm 1 and Figure 1II). The expression pattern over the interval $[t_0, t_{N-1}]$ is partitioned into R equally spaced intervals, denoted as $t_{1,N} = \frac{t_{N-1}-t_0}{R}$. In each interval $[rt_{1,N}, (r+1)t_{1,N})$ for $r \in 0, 1, ..., R-1$, we calculate the maximum and minimum values and compare them to the steady-state measurement at t_0. If the absolute difference between the maximum or minimum exceeds the predefined threshold T at time point t_j, we include the gene in $G(t_j)$ for all $t \geq t_j$.

Algorithm 1. Gene-expression analysis

```
procedure DETECTION(T, R)
    G(t') ← {}, t_0 ≤ t' ≤ t_{N-1}
    for i ∈ {0, 1, 2, ..., M − 1} do
        for r ∈ {0, 1, 2, ..., R − 1} do
            m ← max{x_i(t_k) | k ∈ [rt_{1,N}, (r + 1)t_{1,N})}
            l ← min{x_i(t_k) | k ∈ [rt_{1,N}, (r + 1)t_{1,N})}
            if |x_i(t_0) − m| ≥ T or |x_i(t_0) − l| ≥ T then
                G(t') ← i ∀t' ≥ t
    return G(t)
```

III. Protein Expression Estimation

In a manner very similar to estimating mRNA expression levels, we estimate protein expression by modeling them as a weighted sum of B-spline basis functions (Equation 6).

$$y_i(t) = \sum_{d=1}^{D} \beta_{id}\phi_d(t) = \begin{bmatrix}\phi_1(t) & \cdots & \phi_D(t)\end{bmatrix} \begin{bmatrix}\beta_{i1} \\ \vdots \\ \beta_{iD}\end{bmatrix} = \boldsymbol{\phi}(t)^T\boldsymbol{\beta_i} \tag{6}$$

We solve Equation 3 by rewriting it as:

$$\boldsymbol{\phi}(t)'^T\boldsymbol{\beta_i} = r_i x_i(t) - \lambda_i^{Prot}\boldsymbol{\phi}(t)^T\boldsymbol{\beta_i}$$

$$0 = -r_i x_i(t) + \boldsymbol{\beta_i}(\boldsymbol{\phi}(t)'^T + \lambda_i^{Prot}\boldsymbol{\phi}(t)^T)$$

$$0 = \mathbf{A}_i \begin{bmatrix}-r_i \\ \boldsymbol{\beta_i}\end{bmatrix}$$

$$\mathbf{A}_i = \begin{bmatrix} x_i(t_0) & \lambda_i^{\text{Prot}}\boldsymbol{\phi}(t_0) + \boldsymbol{\phi}'(t_0) \\ x_i(t_1) & \lambda_i^{\text{Prot}}\boldsymbol{\phi}(t_1) + \boldsymbol{\phi}'(t_1) \\ \vdots & \vdots \\ x_i(t_{N-1}) & \lambda_i^{\text{Prot}}\boldsymbol{\phi}(t_{N-1}) + \boldsymbol{\phi}'(t_{N-1}) \end{bmatrix}$$

We can then state this problem as another regularized L_2 norm minimization problem for every gene i

$$\min_{\beta_i} \left\| A_i \begin{bmatrix}-r_i \\ \boldsymbol{\beta_i}\end{bmatrix} \right\|_2 + \gamma_\beta \boldsymbol{\beta}_i^T \mathbf{K} \boldsymbol{\beta}_i \tag{7}$$

in which $\boldsymbol{K}$ is defined equivalently as in Equation 5. Furthermore, we add the constraint that $\boldsymbol{\phi}'^T\boldsymbol{\beta_i} \geq 0$.

IV. Estimating Gene Regulators Coefficients

The final step of the inference pipeline involves estimating the desired weights of Equation 2 for every gene. These gene-specific weights express which protein products regulate a gene and to what extent. To infer them, we solve Equation 1 by transforming it into another optimization problem. To begin with, we can rewrite $\tau_i f_i(Y_{G(t_j)})$ in Equation 1 as follows:

$$\tau_i f_i(Y_{G(t)}) = \frac{\tau_i a_{i0} + \sum_{j=1}^{W(t)} \tau_i a_{ij} \prod_{k\in S_{ij}(t)} y_k(t)}{1 + \sum_{j=1}^{W(t)} b_{ij} \prod_{k\in S_{ij}(t)} y_k(t)}$$

$$= \frac{\begin{bmatrix}1 & \prod_{k\in S_{i1}(t_j)} y_k(t_j) & \cdots & \prod_{k\in S_{iW(t_j)}(t_j)} y_k(t_j)\end{bmatrix} \begin{bmatrix}\alpha_{i0}\tau_i \\ \vdots \\ \alpha_{iW(t_j)}\tau_i\end{bmatrix}}{\begin{bmatrix}1 & \prod_{k\in S_{i1}(t_j)} y_k(t_j) & \cdots & \prod_{k\in S_{iW(t_j)}(t_j)} y_k(t_j)\end{bmatrix} \begin{bmatrix}1 \\ b_{i1} \\ \vdots \\ b_{iW(t_j)}\end{bmatrix}}$$

$$= \frac{\boldsymbol{p}_i(t_j)^T\boldsymbol{a}_i}{\boldsymbol{p}_i(t_j)^T\boldsymbol{b}_i}$$

Note that we have absorbed the term τ_i in $\boldsymbol{a}_i$ and thereby do not need any estimate for τ_i. Equation 1 can then be rewritten for every discrete time point t_j as

$$\frac{dx_i(t_j)}{dt}\Big|_{t=t_j} = \frac{\boldsymbol{p}_i(t_j)^T\boldsymbol{a}_i}{\boldsymbol{p}_i(t_j)^T\boldsymbol{b}_i} - \lambda_i^{RNA}x_i(t_j)$$

$$0 = \frac{\boldsymbol{p}_i(t_j)^T\boldsymbol{a}_i}{\boldsymbol{p}_i(t_j)^T\boldsymbol{b}_i} - \lambda_i^{RNA}x_i(t_j) - \frac{dx_i(t_j)}{dt}\Big|_{t=t_j}$$

$$0 = \boldsymbol{p}_i(t_j)^T\boldsymbol{a}_i - (\lambda_i^{RNA}x_i(t_j) + \frac{dx_i(t_j)}{dt}\Big|_{t=t_j})\boldsymbol{p}_i(t_j)^T\boldsymbol{b}_i$$

We can now remove the fractional nature of the differential equation and formulate the following framework to derive the $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ parameters by solving

$$\min_{\boldsymbol{a}_i,\boldsymbol{b}_i} \left\| \sum_{j=0}^{N-1} (\boldsymbol{p}_i(t_j)^T\boldsymbol{a}_i - (\lambda_i^{RNA}x_i(t_j) + \frac{dx_i(t_j)}{dt}\Big|_{t=t_j})\boldsymbol{p}_i(t_j)^T\boldsymbol{b}_i) \right\|_2$$

$$+\gamma_{i1}\left\|\boldsymbol{b}_i\right\|_2 + \gamma_{i2}\left\|\boldsymbol{b}_i\right\|$$

s.t.

$$0 \leq \boldsymbol{a}_i \leq \boldsymbol{b}_i \quad \boldsymbol{b}_i(0) = 1$$

We obtain the hyperparameters γ_{i1} and γ_{i2} that minimize the objective function the best by parsing over 20 logarithmically scaled values in the range $[-5, 5]$.

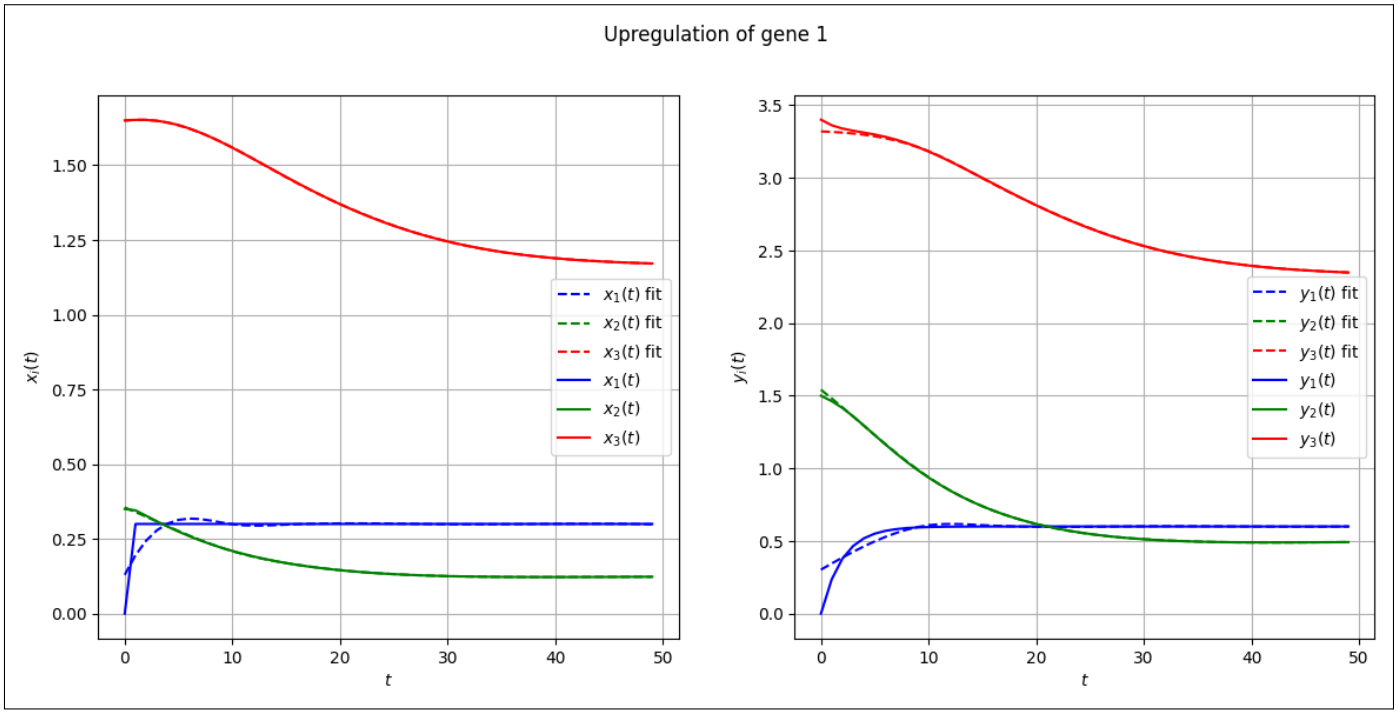


Fig. 2. mRNA expression (left) and protein expression (right) for the 3-gene system 3-gene system as described by the synthetic model in Appendix A. We sampled 50 equally spaced data points and manually set gene 1 to 0.3 after $t = 1$. The pipeline has trouble inferring the expression pattern of gene 1 due to its sudden change in expression level.

4. ARTIFICIAL EXPERIMENT

To assess the pipeline's performance in recovering a dynamical system of gene-gene interactions, we generated artificial data using the synthetic model described in Appendix A, adapted from [5]. Gene expression data were sampled for three genes at 50 time points. One gene was manually set to increase to 0.3 to simulate a change in gene expression in response to an external factor (Figure 2). The advantage of this artificial experiment lies in having access to the ground truth.

The three steps of the inference pipeline show reasonable performance, as observed qualitatively from the fitted expression patterns (Figure 2). Following the approach of the original authors [5], we evaluate the model's effectiveness by comparing the inferred coefficients to the ground truth. The parameter values for $\boldsymbol{a_3}$ and $\boldsymbol{b_3}$ in the ground truth are given as:

$$\boldsymbol{a_3} = \begin{bmatrix} 0.1 & 0 & 0.1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{b_3} = \begin{bmatrix} 1 & 0 & 0.1 & 0.1 & 0 & 0 & 0 \end{bmatrix}$$

Our pipeline at best infers

$$\boldsymbol{a_3'} = \begin{bmatrix} 0.094 & 0. & 0.079 & 0. & 0. & 0. & 0. \end{bmatrix}$$

$$\boldsymbol{b_3'} = \begin{bmatrix} 1. & 0. & 0.025 & 0. & 0. & 0.034 & 0.01 \end{bmatrix}$$

We observe that the model struggles to fully recover the system. Our suspicion is that the artificial sharp peak is too difficult to interpolate, as can be observed qualitatively (Figure 2) and hence the downstream steps in the inference pipeline suffer from this inaccurate capture of the gene expression dynamics. Adding more data, i.e. sampling more time steps and trying different starting points, did not improve inference significantly. To delve deeper into how far off the assemble of inferred coefficients is, we re-plotted the system with the inferred weights (figure 3). Once again, we see that the pipeline can capture the smooth gene expression patterns and overall dynamics of the system quite well but suffers for genes that have sudden changes in their expression. This limitation of the model was kept in mind during the rest of the experiments.

5. REAL-WORLD APPLICATION: MURINE EMBRYONIC DEVELOPMENT DATA

A. Data preprocessing

To evaluate the model's performance on real biological data and assess its ability to infer gene-gene interactions during cell differentiation, we utilized single-cell RNA sequencing data from 116,312 mouse cells sampled at 9 time points by *Pijuan-Sala et al.* [6]. The data is accessible at Atlas accession E-MTAB-6967[1]. Expression levels were measured at six-hour intervals post-fertilization between stages E6.5 and E8.5 (Figures 4A & 4B). For the analysis, we extracted RNA counts (representing gene expression) of key genes in 6 cell types during embryonic development. The selected genes are Nanog, Pou5f1 (Oct4), Sox2, Tbx3, Klf2, and Nodal. These genes are known to be key regulators in various developmental stages, exhibiting high fluctuations in their expression levels over time [15]. For all 37 cell types, we processed the raw counts to obtain a read count per gene per embryonic developmental stage, keeping track of the number of cells per cell type expressing a specific gene at a given stage. Subsequently, we corrected for the number of cells contributing to the read count and performed min-max

[1] https://www.ebi.ac.uk/biostudies/arrayexpress/studies/E-MTAB-6967

normalization across all counts for the 6 genes at 9 discrete time points.

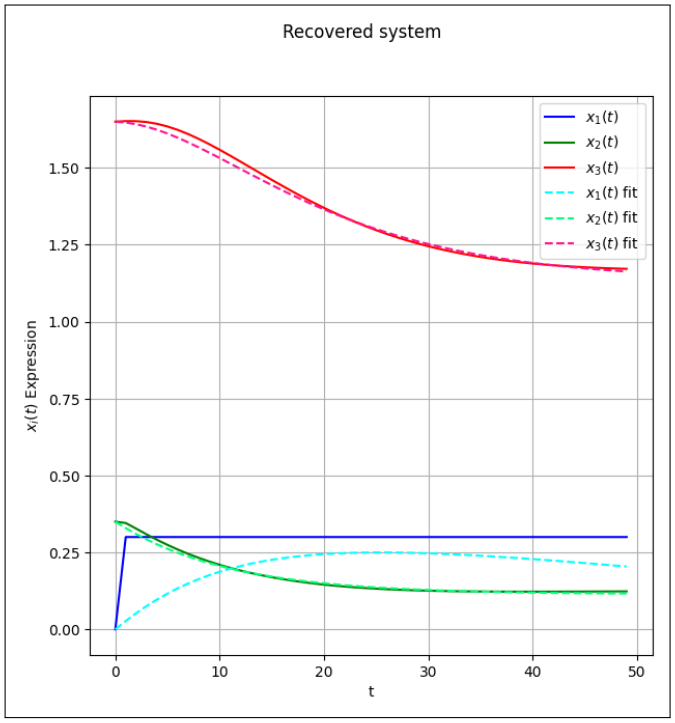


Fig. 3. The inferred system $x_i(t)fit$ vs. the ground truth $x_i(t)$. After running the inference pipeline, we substituted the original weights for our inferred weights in the differential equations (Appendix A).

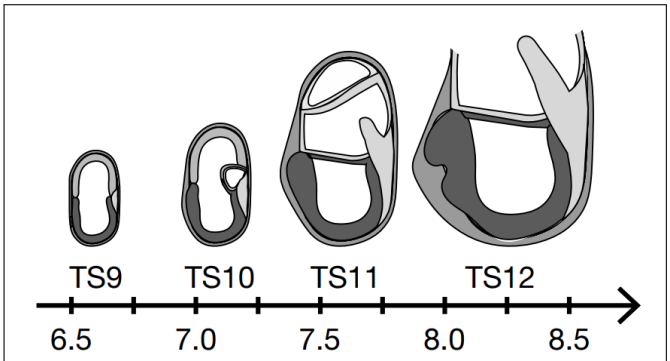
B. Experiment

For all cell types, we ran the inference pipeline and plotted the corresponding network (see Appendix A for links to all inferred weights and networks). Here, we highlight the inference pipeline for two cell types: mesenchyme and visceral endoderm.

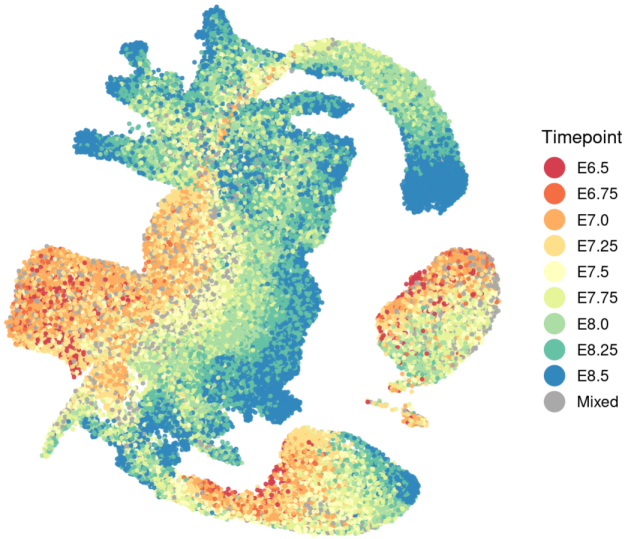
Figure 5 shows the RNA gene-expression patterns interpolated and analyzed for our out-of-steady state genes. We visually observe the benefit of the B-spline gene-expression function as it mitigates modeling the noisy pattern while following the general trend of the genes.

The corresponding networks for these cell types (Figure 6) have relatively low number of interactions inferred, showing the sparseness that the inference method favors. The thickness of the arrows represent the relative amount of regulation compared to the other arrows. Green arrows represent activation, red arrows inhibition.

Our pipeline identified the *Pou5f1-Sox2* regulatory complex that activates *Nanog* in the visceral endoderm [16]. As for the other interactions, determining how accurately they represent true gene-gene interactions in murine embryonic cells is challenging. These interactions might indirectly demonstrate connectivity through other regulators, yet there is no literature corroborating these findings. The mesenchyme network correctly infers the positive self-regulation of key regulators *Nanog* and *Pou5f1*. However, the activating activity of *Tbx3* is mistakenly inferred. Additionally, *Sox2* is erroneously downregulated by *Pou5f1*, and assessing the correctness of other regulators is equally challenging. Overall, the results appear less promising than anticipated,



(a) Murine embryonic developmental stages at which scRNA-seq data was sampled. Numbers indicate days after fertilization. TS stands for Theiler Stages. They form a set of developmental stages used to describe the embryonic development of mice. Taken and adapted from figure 1A in [6].



(b) UMAP of scRNA-seq dataset sampled at different points over time. Taken and adapted from data exploration tool in [6]. The color gradients from red to blue in different directions represent different cell specializations during differentiation. We chose a subset of these paths for our analysis.

Fig. 4. Mouse embryonic development sampled every 6 hours from E6.5 until E8.5 post-fertilization.

particularly for other cell types like endothelium or erythroid cells. Many false positives and false negative interactions are inferred (see Appendix A).

6. DISCUSSION & CONCLUSION

In this project, we have implemented a gene regulatory network inference pipeline based on the biophysics model described by *Zamanighomi et al.* and *Bintu et al.* [5, 9]. First, we simplified the model for it to be able to capture second-order gene-gene regulatory behavior. For artificial data, we were successful up to a certain extent. The tumultuous gene expression patterns were difficult to capture, which hampered network inference later on. As the authors of [5] suggest, modeling smooth perturbations in a gene network should improve matters. We tried this but did not see an overall improvement in the inferred coefficients (Appendix 7). Where one coefficient would improve, another one would be less well characterized. We also tried adding multiple perturbations (Appendix 7) but, again, did not see an overall

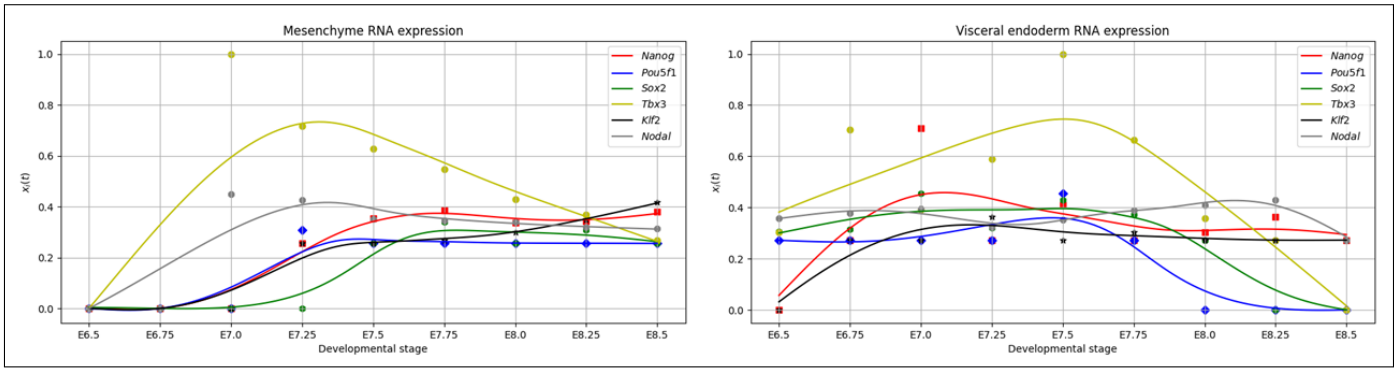


Fig. 5. Min-max normalized mRNA expression for mesenchyme (left) and visceral endoderm (right).

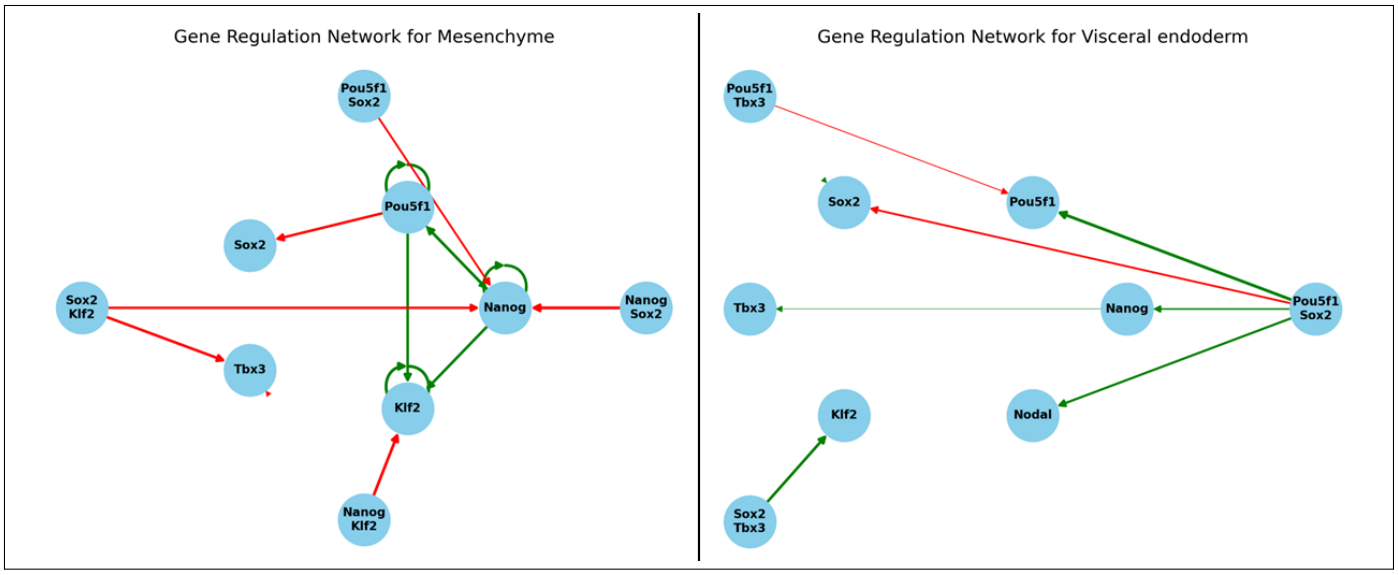


Fig. 6. The final gene-gene interaction networks for mesenchyme and visceral endoderm cell-types left and right, respectively. The relative size of the arrows represent the extent of regulation (i.e. activation or inhibition). Note for example that there is a green activating arrow from $\{Pou5f1, Tbx3\}$ to $Sox2$ but barely see-able due to the low level of activation. All other networks can be found online in the links in the appendix (7).

improvement of the inference. The search space is most probably multimodal and we are stuck in a local minimum during optimization. Trying the different optimization algorithms in `scipy` [14] did not improve matters. Nevertheless, even though the global minimum could not be found, we do capture the overall dynamics of the system as can be observed in our reconstruction plots (Figure 3 and Appendix A).

Secondly, we attempted, for the first time, to infer gene-gene interactions during embryonic mouse development from scRNA sequencing data using this model. As described previously, the framework was designed initially to infer a gene-gene interaction network from gene expression measurements that are out-of-steady state. Embryonic cell expression is constantly activated and repressed due to external cues and cell-cell communication [8]. Hence, we ought it probable that the simplified model could capture gene-gene interactions in embryonic gene expression data. For two cell types, mesenchyme and visceral endoderm, we correctly established a few regulatory interactions but the majority of interactions is could not be found back in literature. This problem is even more prominent in other cell types (see links to data in appendix (A)). Errors in inference might be due to the sparsely sampled gene expression and the inherent noise of the measurements. Furthermore, we have already observed in the artificial experiments that our model suffers especially from chaotic gene expression patterns. In addition, we now set unknown parameters (e.g. r_i, λ_i^{RNA}, etc.) to value 1 out of convenience and due to a lack of information on these but more accurate knowledge on these parameters should improve inference as well. Lastly, the search space might contain too many optima, causing the minimization problem to be stuck locally and thereby wrongfully inferring the weights. Nonetheless, we successfully extracted meaningful signals, providing a solid foundation for future enhancements and advancements.

One possible improvement in the inference pipeline would be to solve Equation 1 through other optimization methods like SINDY-Pi [17]. *Mangan et al.* even already went this far by using the implicit-SINDy algorithm directly on the mRNA expression patterns to infer gene-gene interactions [18]. A next step in this line of research would be to see if the implicit-SINDy algorithm can also infer the coefficients in the fractional term of Equation 1. If this inference is successful for the simple model, including extra regulators like microRNAs could then lead to

more improvements. Given all this, the biophysics model still has a lot of potential and with the increase in the quantities of data nowadays, it might elucidate new gene-gene interactions in the future.

Data & source code. All data and code needed for reproducing the experiments and creating figures shown in this report are available online at https://github.com/MaEduard/network_inference.

Acknowledgments. Many thanks to Prof. Dr. Ir. M.J.T. Reinders and Dr. A. Villegas Morcillo for guiding me throughout this honours project. Due to your guidance, constructive feedback, and fruitful discussions, I experienced for the first time what it is like to get my hands dirty in research. Furthermore, thank you Gerard Bouland for helping me in the data acquisition process.

Disclosures. The format was adapted and taken from the (https://www.overleaf.com/latex/templates/length-check-latex-template-for-preparing-an-article-for-submission-to-optica-publishing-group-journals-ao-jocn-josa-a-josa-b-ol-optica/xjznbfqymcxm).

7. APPENDIX

A. synthetic Model

$$
\begin{aligned}
\frac{dx_1}{dt} &= \frac{0.1 + 0.05y_1y_2 + 0.025y_1y_3}{1 + 0.1y_1 + 10y_3 + 0.05y_1y_2 + 0.025y_1y_3} - 0.1x_1 \\
\frac{dx_2}{dt} &= \frac{0.1 + 0.1y_1 + 0.1y_1y_2}{1 + 0.1y_1 + 0.1y_1y_2 + 10y_1y_3} - 0.1x_2 \\
\frac{dx_3}{dt} &= \frac{0.1 + 0.1y_2}{1 + 0.1y_2 + 0.1y_3} - 0.1x_3 \\
\frac{dy_1}{dt} &= x_1 - 0.5y_1 \\
\frac{dy_2}{dt} &= 2x_2 - 0.5y_2 \\
\frac{dy_3}{dt} &= x_3 - 0.5y_3
\end{aligned}
$$

RNA expression inference embryonic data. All synthetic and biological data can be found back here: https://github.com/MaEduard/network_inference/tree/main/data. All figures, including plots not shown in the report, can be found here:

1. rna-expression: https://github.com/MaEduard/network_inference/tree/main/src/rna_expression_bio
2. protein expression: https://github.com/MaEduard/network_inference/tree/main/src/protein_expression_bio
3. network plots: https://github.com/MaEduard/network_inference/tree/main/src/network_plots

FULL REFERENCES

1. R. Zhang, S. Shen, Y. Wei, *et al.*, "A large-scale genome-wide gene-gene interaction study of lung cancer susceptibility in europeans with a trans-ethnic validation in asians," J. Thorac. Oncol. **17**, 974–990 (2022).
2. H. J. Cordell, "Detecting gene–gene interactions that underlie human diseases," Nat. Rev. Genet. **10**, 392–404 (2009).
3. M. Marku and V. Pancaldi, "From time-series transcriptomics to gene regulatory networks: A review on inference methods," PLOS Comput. Biol. **19**, e1011254 (2023).
4. B. Schmidt and A. Hildebrandt, "Next-generation sequencing: big data meets high performance computing," Drug discovery today **22**, 712–717 (2017).
5. M. Zamanighomi, M. Zamanian, M. Kimber, and Z. Wang, "Gene regulatory network inference from perturbed time-series expression data via ordered dynamical expansion of non-steady state actors," IEEE/ACM transactions on computational biology bioinformatics **15**, 1093–1106 (2015).
6. B. Pijuan-Sala, J. A. Griffiths, C. Guibentif, *et al.*, "A single-cell molecular map of mouse gastrulation and early organogenesis," Nature **566**, 490–495 (2019).
7. M. Lange, A. Granados, S. VijayKumar, *et al.*, "Zebrahub–multimodal zebrafish developmental atlas reveals the state-transition dynamics of late-vertebrate pluripotent axial progenitors," bioRxiv pp. 2023–03 (2023).
8. A. Meister, Y. H. Li, B. Choi, and W. H. Wong, "Learning a nonlinear dynamical system model of gene regulation: A perturbed steady-state approach," The Ann. Appl. Stat. pp. 1311–1333 (2013).
9. L. Bintu, N. E. Buchler, H. G. Garcia, *et al.*, "Transcriptional regulation by the numbers: models," Curr. opinion genetics & development **15**, 116–124 (2005).
10. B. Alberts, *Molecular biology of the cell* (Garland science, 2017).
11. B. E. Slatko, A. F. Gardner, and F. M. Ausubel, "Overview of next-generation sequencing technologies," Curr. protocols molecular biology **122**, e59 (2018).
12. D. Kraft, "A software package for sequential quadratic programming," Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt (1988).
13. M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing. i. theory," IEEE transactions on signal processing **41**, 821–833 (1993).
14. P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, "Scipy 1.0: fundamental algorithms for scientific computing in python," Nat. methods **17**, 261–272 (2020).
15. B. K. Hall, *Evolutionary developmental biology* (Springer Science & Business Media, 2012).
16. Y.-H. Loh, Q. Wu, J.-L. Chew, *et al.*, "The oct4 and nanog transcription network regulates pluripotency in mouse embryonic stem cells," Nat. genetics **38**, 431–440 (2006).
17. K. Kaheman, J. N. Kutz, and S. L. Brunton, "Sindy-pi: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics," Proc. Royal Soc. A **476**, 20200279 (2020).
18. N. M. Mangan, S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Inferring biological networks by sparse identification of nonlinear dynamics," IEEE Trans. on Mol. Biol. Multi-Scale Commun. **2**, 52–63 (2016).