Linear system identification from ensemble snapshot observations

Atte Aalto and Jorge Gonçalves

***Abstract*— Developments in transcriptomics techniques have caused a large demand in tailored computational methods for modelling gene expression dynamics from experimental data. Recently, so-called single-cell experiments have revolutionised genetic studies. These experiments yield gene expression data in single cell resolution for a large number of cells at a time. However, the cells are destroyed in the measurement process, and so the data consist of snapshots of an ensemble evolving over time, instead of time series. The problem studied in this article is how such data can be used in modelling gene regulatory dynamics. Two different paradigms are studied for linear system identification. The first is based on tracking the evolution of the distribution of cells over time. The second is based on the so-called pseudotime concept, identifying a common trajectory through the state space, along which cells propagate with different rates. Therefore, at any given time, the population contains cells in different stages of the trajectory. Resulting methods are compared in numerical experiments.**

I. Introduction

Introduction of high–throughput sequencing technologies has caused an increase in produced gene expression data, and even time series data have become widely available. This has raised computational modelling of genetic systems to the pinnacle of today's research in biology. The cost of collecting data is still very high compared to mechanical or electrical systems, for example, and so the gene expression time series tend to be short in length and the sampling frequency low, which has created a demand for tailored methods taking into account the limitations in the data.

Recent years have witnessed another revolution in sequencing technologies. With so-called single-cell techniques, it is possible to obtain gene expression measurements at the level of one cell instead of a population average obtained by traditional batch techniques. Unfortunately, the cell is destroyed in the measurement process, and therefore it is possible to get only one measurement per cell — albeit from a large number of cells at a time. The amount of data is orders of magnitude larger than with batch experiments, but the obtained ensemble snapshot data call for new modelling approaches. In this paper, we consider this problem from the point of view of linear system identification. Although simplistic, the goal of this work is to obtain evidence on the suitability of the overall strategies for tackling the problem.

AA was supported by ERANET for Systems Biology ERASysApp and Fonds National de la Recherche Luxembourg, project CropClock, grant reference INTER/SYSAPP/14/02, and University of Luxembourg Internal research projects PPPD and OptBioSys. JG was partly supported by the 111 Project on Computational Intelligence and Intelligent Control under Grant B18024.

Both authors are with Luxembourg Centre for Systems Biomedicine; University of Luxembourg; 6 Avenue du Swing; 4367 Belvaux; Luxembourg. (email: atte.aalto@uni.lu and jorge.goncalves@uni.lu)

A typical single-cell experiment is carried out as follows. The considered cell population consisting of $N = \sum_{j=0}^{m} N_j$ cells normally originates from a clonal population so that the cells can be expected to behave similarly. At time $T_0 = 0$, a sub-population of N_0 cells is measured. At the same time, remaining cells are perturbed somehow, depending on the experiment, for example by introducing a drug or some other stimulant. At later times T_j, sub-populations consisting of N_j cells are measured. In the end, the measurement data consist of $m+1$ snapshot observations of ensembles, $Y = \{Y_0, Y_1, ..., Y_m\}$, where $Y_j = \left[y_1^{(j)}, ..., y_{N_j}^{(j)}\right] \in \mathbb{R}^{n \times N_j}$. The vector $y_k^{(j)} \in \mathbb{R}^n$ consists of gene expression levels of n (interesting) genes in the k^{th} cell measured at time T_j. For a review on single-cell experimental techniques and a discussion on their potential, we refer to [1].

In this paper, we consider linear system identification from data mimicking a single-cell experiment. Assume that the gene expression dynamics of the cell $k \in \{1, ..., N_j\}$ in the sub-population $j \in \{0, ..., m\}$ are governed by

$$\mathrm{d}x_k^{(j)} = \gamma_k^{(j)} A x_k^{(j)} \mathrm{d}t + \mathrm{d}u_k^{(j)}, \qquad x_k^{(j)}(0) \sim P_0 \tag{1}$$

where A is a sparse matrix (since the dynamics of one gene are known to be influenced by only few other genes), and $u_k^{(j)}$ is a noise process modelled as Brownian motion. The time-scaling constant $\gamma_k^{(j)} > 0$ models the development rate of the cell, which varies from cell to cell. The initial state is a random variable with probability distribution P_0. The measurement obtained from this cell is

$$y_k^{(j)} = x_k^{(j)}(T_j) + v_k^{(j)},$$

where $v_k^{(j)}$ is measurement noise, and T_j is the measurement time. The assumption of a full-state measurement is of course a simplification, but it is a rather typical one in genetic applications, made to avoid overfitting.

The problem is to estimate the (sparse) matrix A from the ensemble snapshot data Y. We introduce two different paradigms for approaching the problem, and develop one method within each paradigm. Firstly, we develop a method based on tracking the propagation of the (probability) distributions of cells over time, and finding a sparse matrix A that produces such propagation. The second paradigm is based on the so-called pseudotime concept [2]–[4]. The underlying idea in this concept is that cell dynamics are not identical through the population, and in particular, some cells develop faster than others. Therefore, the distribution of measurements at time T_j contains information from different developmental stages. Pseudotime refers to the stage of the cell in the development process. In the example (1), the

pseudotime of the measurement of cell k measured at time T_j roughly corresponds to $\gamma_k^{(j)} T_j$. Pseudotime methods infer the developmental stage of each measured cell. In [5] we developed a method for estimating the zero structure of the dynamics matrix A from time series data. Here this method is modified to include an additional estimator for the pseudotime for each measurement, which is carried out simultaneously with the zero structure inference. In the context of gene expression modelling, the zero structure of A can be interpreted as the gene regulatory network (GRN). GRN inference is one of the cornerstone problems studied in systems biology [5]–[10]. The two developed methods are compared to the method presented in [5] applied on the time series data consisting of the averages of sub-populations Y_j. This corresponds to a traditional gene expression measurement producing a short time series.

Introduction of single-cell sequencing techniques has lead to emergence of methods analysing the resulting data. Methods that infer cell dynamics from such data include [11]–[13]. The first two works are concerned with estimating the state distribution from incomplete measurements. The article [13] introduces a method for obtaining distributions of unknown parameters in a chosen dynamical model from the measurement distributions. Optimal mass transport has been applied to single-cell data in [14] for reconstructing cell trajectories. GRN inference from single-cell data has been discussed in [7], [8]. Inference is done, for example, using gene expression correlations [9], or by considering stationary distributions arising from a mechanistic model [10].

II. Methods

The three methods in the comparison are presented in this section. The first, distribution-based method is completely new, and the second, pseudotime-based method is a modification of our earlier method using time series data [5]. The third method is our original method (without the modification) applied on the population average, which corresponds to data obtained from a classical batch experiment. The distribution-based method is estimating the full matrix A, whereas the method in [5] is developed for inferring the GRN, that is, the zero structure of A. In Section III, the methods are compared in the GRN inference task.

A. Distribution-based method

The dynamics equation (1) defines the cell trajectory as a stochastic process (if also the development rate $\gamma_k^{(j)}$ is a random variable). At time T_j the cell state has a certain probability distribution P_j (finite-dimensional distribution of the stochastic process), and the measurements Y_j are regarded as samples drawn from this distribution. The idea is to find a matrix A, such that the pushforward measure $e^{A(T_1-T_0)}P_0$ would be close to P_1, and similarly for all $j \in 1,...,m$, the pushforward measure $e^{A(T_j-T_{j-1})}P_{j-1}$ should be close to P_j. The "closeness" is measured by the Jensen–Shannon divergence between the two distributions [15] (see Remark 1). The Jensen–Shannon divergence is defined through the Kullback–Leibler divergence as

$$\mathrm{JS}(p \,\|\, q) = \frac{1}{2}\mathrm{KL}(p \,\|\, m) + \frac{1}{2}\mathrm{KL}(q \,\|\, m)$$

where $m = \frac{1}{2}(p+q)$. Recalling the definition of the Kullback–Leibler divergence, the Jensen–Shannon divergence can be expressed as

$$\begin{aligned}\mathrm{JS}(p \,\|\, q) = &\frac{1}{2}\int \log(p(x))p(x)\mathrm{d}x \\ &+ \frac{1}{2}\int \log(q(x))q(x)\mathrm{d}x - \int \log(m(x))m(x)\mathrm{d}x. \quad (2)\end{aligned}$$

As opposed to the Kullback–Leibler divergence, the Jensen–Shannon divergence is symmetric with respect to p and q. In addition, there is no absolute continuity requirement between the measures correponding to p and q. The continuity requirement for Kullback–Leibler divergence is always satisfied, since $m(x) = 0$ implies $p(x) = 0$ and $q(x) = 0$.

The identification task can then be formulated as an optimisation problem

$$\min_A \; C(A) + \sum_{j=1}^{m} \mathrm{JS}\left(e^{A(T_j-T_{j-1})}P_{j-1} \,\|\, P_j\right) \quad (3)$$

where $C(A)$ is some sparsity promoting regulariser, for example $C(A) = \lambda \sum_{i,j} |A_{i,j}|$ is used in our numerical experiment, corresponding to the well-known Lasso approach [16].

The optimisation problem in (3) is defined for full distributions P_j, but the data consist of samples from those distributions. Therefore integrals of the form $\int \log(p(x))p(x)\mathrm{d}x$ have to be approximated using samples $x_1,...,x_L$ drawn from p. Two different approximations for the distribution p are used. The latter $p(x)$ in the integral is approximated by a sum of Dirac delta distributions at the sample points, transforming the integral into a sum (see Remark 2)

$$\int \log(p(x))p(x)\mathrm{d}x \approx \frac{1}{L}\sum_{j=1}^{L} \log(p(x_j)). \quad (4)$$

The remaining $p(x)$ is approximated with a Gaussian mixture

$$p(x) \approx \frac{1}{L(2\pi q)^{n/2}} \sum_{j=1}^{L} \exp\left(-\frac{|x-x_j|^2}{2q}\right) \quad (5)$$

where q is a design parameter. Inserting this into (4) gives

$$\begin{aligned}&\int \log(p(x))p(x)\mathrm{d}x \\ &\approx \frac{1}{L}\sum_{j=1}^{L} \log\left(\sum_{\substack{k=1\\k\neq j}}^{L} \exp\left(-\frac{|x_k-x_j|^2}{2q}\right)\right) + C \quad (6)\end{aligned}$$

where $C = -\log(L(2\pi q)^{n/2})$ and the $k = j$ term has been excluded from the sum, since otherwise the method seemed to give too little weight to measurements on the outskirts of the distribution.

Practical implementation of the method is sketched in Algorithm 1, where the optimisation problem (3) with approximation (6) is solved using simulated annealing. The

approximated Jensen–Shannon divergence $\widetilde{\mathrm{JS}}(X\|Y)$ for $X \in \mathbb{R}^{n\times m_X}$ and $Y \in \mathbb{R}^{n\times m_Y}$ is computed as follows. The first term in (2) is computed by inserting X into (6), the second term by inserting Y, and the last term by inserting $[X, Y]$.

The parameter q in (5) and (6) was chosen differently when computing different terms of the sum in (3). For the j^{th} term in the sum, it was chosen as one tenth of the average of the values $|y_i^{(j)} - y_k^{(j)}|^2$ for $i, k \in \{1, ..., N_j\}$ and $i \neq k$.

for $i = 1, ..., n_{its}$ **do**
  Draw $\hat{A} = A^{(i-1)} + \epsilon_i \cdot \mathrm{randn}(n, n)$;
  Set $J = C(\hat{A})$;
  for *j=1,...,m* **do**
    Compute $X_j = e^{\hat{A}(T_j - T_{j-1})} Y_{j-1}$;
    Set $J = J + \widetilde{\mathrm{JS}}(X_j\|Y_j)$;
  end
  if $\exp\big((J_{\text{old}} - J)/\mathrm{Temp}_i\big) > \mathrm{rand}$ **then**
    set $J_{\text{old}} = J$;
    set $A^{(i)} = \hat{A}$;
  else
    set $A^{(i)} = A^{(i-1)}$;
  end
end

Algorithm 1: The distribution-based method. The simulated annealing temperature Temp_i and step size ϵ_i decrease as the iterations proceed. Here rand and randn denote random variables from the uniform distribution $U(0, 1)$ and the normal distribution $N(0, 1)$, respectively.

Remark 1. In our experiments, also $\mathrm{KL}(p\|q) + \mathrm{KL}(q\|p)$ was tried as a distance measure between distributions, but the Jensen–Shannon divergence seemed to produce slightly better results.

Remark 2. The approximation (4) can also be obtained by immediately replacing $p(x)$ by the Gaussian mixture (5), and then approximating the integral using Gauss–Hermite quadrature with only one sample point per one Gaussian distribution in the mixture sum. A better result could perhaps be obtained by using more quadrature sampling points, but this would slow down the computations somewhat, in particular if the dimension n is big.

B. Simultaneous estimation of pseudotime and the matrix A

A method for estimating the zero structure of the matrix A from time series data was developed in [5]. The method is based on Bayesian analysis and MCMC sampling. The method also samples the continuous-time trajectory x underlying the sparsely sampled time series data. Similarly, in the pseudotime concept, it is assumed that the measurements are produced by a continuous trajectory x, along which the cells propagate with different rates. In this section, a variant of this method will be developed, where also the pseudotimes related to the measurements are estimated simultaneously. To put briefly, the modification made to the method in [5] is that an additional MCMC sampler is constructed for the pseudotime. In this case, the measurement distribution, given the continuous trajectory x, is $y_k^{(j)} \sim N(x(\tau_k^{(j)}), R)$ where $\tau_k^{(j)}$ is the pseudotime corresponding to the measurement $y_k^{(j)}$. In [5] the measurement time was fixed and the measurements readily formed a time series, and the measurement distribution was $y_j \sim N(x(t_j), R)$ where t_j was the measurement time of y_j.

In this method, an indicator variable is introduced for the zero structure of the matrix A. That is, the element $A_{i,j}$ is represented as a product $A_{i,j} = S_{i,j}H_{i,j}$ where $S_{i,j} \in \{0, 1\}$ is an indicator variable indicating whether the element (i, j) is zero or not, and $H_{i,j} \in \mathbb{R}$ is the magnitude variable. The object of interest is then the posterior distribution of the indicator variable S, given the data $Y = [Y_0, ..., Y_m]$, and the corresponding measurement times $T = \{T_0, ..., T_m\}$:

$$\begin{aligned} p(S|Y, T) &\propto p(Y|S, T)p(S) \\ &= p(S) \iiint p(Y, x, H, \tau|S, T)\mathrm{d}x\,\mathrm{d}H\,\mathrm{d}\tau \\ &= p(S) \iiint p(Y|x, \tau)p(x|S, H)p(\tau|T)p(H)\mathrm{d}x\,\mathrm{d}H\,\mathrm{d}\tau \end{aligned}$$

where we have first used the Bayes' law, then introduced the latent variables x, τ, and H, where $\tau = \{\tau_1^{(0)}, ..., \tau_{N_0}^{(0)}, ..., \tau_1^{(m)}, ..., \tau_{N_m}^{(m)}\}$ is the pseudotime vector, H is the magnitude variable, and x is the continuous trajectory. Finally, the probability chain rule is applied to obtain known distributions. The measurement model is

$$p(Y|x, \tau) = \prod_{\substack{j=0,\dots,m \\ k=1,\dots,N_j}} N\left(y_k^{(j)}; x(\tau_k^{(j)}), R\right),$$

that is, it is assumed that the measurements are obtained from the same trajectory at different developmental stages, which is not the same as the true measurement time.

The integral with respect to the magnitude variable $H \in \mathbb{R}^{n\times n}$ is possible to carry out analytically (it is done in [5]), assuming that the rows of H are normally distributed $H_i \sim N(0, M_i)$, and independent. A time interval $[\underline{T}, \overline{T}]$ is defined for the continuous trajectory. The pseudotimes should be contained in this interval. The integral is

$$\begin{aligned} &\int p(x|S, H)p(H)\mathrm{d}H \\ &\propto \prod_{i=1}^{n} \frac{\exp(\Phi_i(x))}{\left|M_i[S_i]^{-1} + \frac{1}{q_i}\mathbb{X}[S_i]\right|^{1/2}\left|M_i[S_i]\right|^{1/2}} \mathcal{W}_Q(\mathrm{d}x) \end{aligned}$$

where the functionals $\Phi_i(x)$ are

$$\begin{aligned} \Phi_i(x) := \frac{1}{2q_i^2}&\left(\int_{\underline{T}}^{\overline{T}} x[S_i]^\top \mathrm{d}x_i\right) \\ &\cdot \left(M_i[S_i]^{-1} + \frac{1}{q_i}\mathbb{X}[S_i]\right)^{-1} \left(\int_{\underline{T}}^{\overline{T}} x[S_i]\mathrm{d}x_i\right), \end{aligned}$$

$\mathcal{W}_Q(\mathrm{d}x)$ is the Wiener measure with incremental covariance matrix $Q = \mathrm{diag}(q_1, ..., q_n)$ corresponding to noise process $u_k^{(j)}$ in (1), and $\mathbb{X} = \int_{\underline{T}}^{\overline{T}} x(t)x(t)^\top \mathrm{d}t$ is the Gramian matrix. The notation $x[S_i]$ where S_i is the i^{th} row of S, means the

subvector of x in $\mathbb{R}^{|S_i|_0}$ that consists of those elements x_j for which $S_{i,j} = 1$, and for a matrix $K \in \mathbb{R}^{n\times n}$, the notation $K[S_i]$ stands for the $|S_i|_0 \times |S_i|_0$ submatrix of K consisting of those rows and columns of K for which $S_{i,j} = 1$.

The integrals with respect to x and τ are carried out by MCMC sampling. The prior for the pseudotime is a normal distribution. For measurement k done at time T_j, we set $p(\tau_k^{(j)}|T) = N(T_j, \sigma_\tau)$ (truncated so that $\tau_k^{(j)} \in [\underline{T}, \overline{T}]$). Also the covariance parameters $Q = \mathrm{diag}(q_1, ..., q_n)$ and $R = \mathrm{diag}(r_1, ..., r_n)$ are sampled, as well as the indicator matrices S, for which the prior is $p(S) \propto \rho^{|S|_0}$ where $\rho \in (0, 1)$ is a parameter controlling the sparsity of the samples. For the average of these samples $S^{(j)}$, it holds that

$$\frac{1}{L}\sum_{j=1}^{L} S^{(j)} \to \mathbb{E}(S|Y, T), \qquad \text{as } L \to \infty$$

and this average is the output of the algorithm. Since S is a Boolean variable, the elements of $\mathbb{E}(S|Y, T)$ are actually the posterior probabilities that the corresponding elements in A are nonzero. The details on the practical implementation of the MCMC sampler as well as details on the computation of the above integrals can be found in [5].

C. Batch average tracking

As opposed to novel single-cell techniques, older batch sequencing techniques are only able to provide measurements from population averages. Corresponding to such setup, the method developed in [5] is also included in the comparison, using time series data (with length $m+1$) obtained from the population means

$$y_j = \frac{1}{N_j}\sum_{k=1}^{N_j} y_k^{(j)}$$

with measurement times T_j, for $j = 0, ..., m$.

III. Numerical experiments

To generate the experimental data, equation (1) was numerically simulated separately for each cell. The used dynamics matrix was

$$A = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & -1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -2 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

corresponding to the gene regulatory network shown in Figure 1. The diagonal elements are chosen so that each column sum is zero. The development rates were drawn from a uniform distribution $\gamma_k^{(j)} \sim U(1, 1.2)$. The driving Brownian motion $u_k^{(j)}$ had incremental covariance $0.01I$.

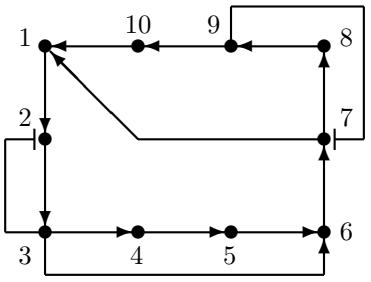

Fig. 1: The gene regulatory network corresponding to the matrix A of the example. Arrows denote positive effects and blunt arrows denote negative effects.

The methods were compared in the task corresponding to gene regulatory network inference, that is, inference of the zero structure of the matrix A. Well-known classifier scores are used in the comparison, namely the area under the receiver operating characteristic curve (AUROC) and the area under the precision–recall curve (AUPR), excluding the diagonal elements. For the computation of the AUROC and AUPR scores, the methods need to rank the potential links (elements in the A matrix) in the order of confidence. For the distribution-based method, the confidence ranking is obtained simply by ordering the elements of A in decreasing order of their absolute values. This comparison is not entirely fair to the distribution-based method, since it is actually estimating the matrix A, rather than the probabilities for the entries being nonzero like the other two methods.

A. Experiment 1

In the first experiment, altogether 445 measurements are collected at eight different times, as described in Table I. The initial distribution is a normal distribution, $P_0 = N(m_0, \Gamma)$ where m_0 was also randomly chosen and $\Gamma = \mathrm{diag}(.1, .05, .16, .2, .11, .19, .18, .07, .11, .09)^2$. The data are visualised in Figure 2 (left) showing first two principal components. From the figure it can be seen that the later measurements are more spread in the direction of the main propagation due to the different development rates.

The distribution-based method was tested with six different values of λ, which is the cost function parameter penalising for the 1-norm of the A-matrix. Similarly, the two other methods were tried with six different values of the sparsity parameter ρ. The resulting AUROC and AUPR values (for four interesting parameter values) are shown in Table II. The pseudotime method shows a more solid performance than the distribution-based method, even obtaining perfect

TABLE I: Measurement times T_j and sizes of measured subpopulations N_j in the different experiments.

Exp. 1	j	0	1	2	3	4	5	6	7
Exp. 2–4a	j	0		1	2		3		4
	T_j	0	0.2	0.5	1.2	2.2	2.95	4	5.2
	N_j	50	59	57	55	54	64	60	46

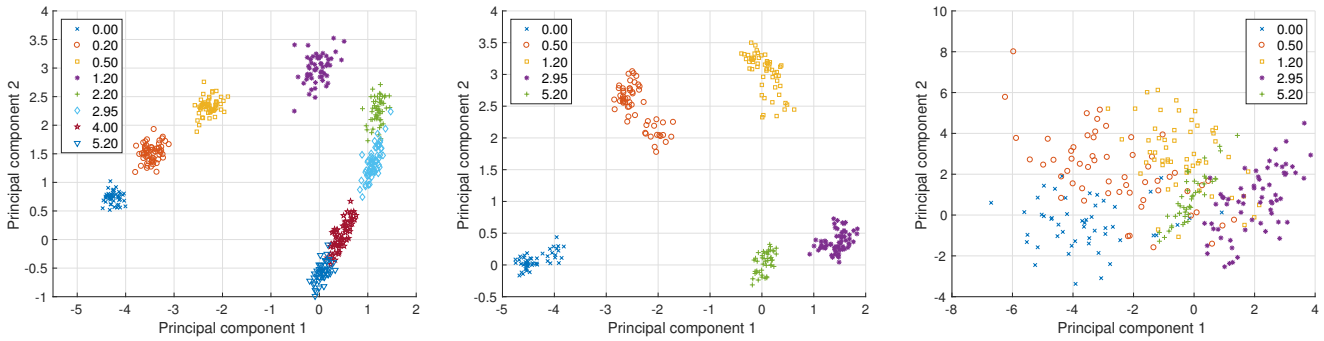

Fig. 2: The first two principal components of the simulated data for experiments 1 and 2 (left), experiment 3 (center), and experiment 4a (right). Each point corresponds to one measured cell. Different measurement times are indicated with different symbols and colours.

reconstruction with $\rho = 0.3$. The batch-method *C* is clearly the weakest, which is not surprising.

B. Experiment 2

In the second experiment, the amount of data was reduced by using only five of the eight populations in Experiment 1 (see Table I and Figure 2 (left)) resulting in 272 measurements. In this experiment, only the values $\lambda = 0.005$ and $\rho = 0.2$ were used. The results are shown in Table II.

Again, the pseudotime method *B* was better than the distribution-based method *A*. However, the pseudotime method seemed to be sensitive to the initialisation of the continuous trajectory x in the sampler, and sometimes the MCMC sampler converged to the neighbourhood of a local maximum of the posterior distribution, which did not yield as good results as those reported in Table II. Since the batch-method *C* only has five data points in this experiment, its performance is clearly worse than in Experiment 1.

C. Experiment 3

In the third experiment, the initial distribution P_0 was a mixture of two Gaussians, so that with probability 0.7, the initial state $x_k^{(j)}(0)$ was drawn from normal distribution $N(m_0, \Gamma)$, and with probability 0.3, the initial state was drawn from $N(m_1, \Gamma)$, where m_0 and m_1 were close to each other. This experiment is simulating heterogeneity in the cell population. The measurement times and population sizes are the same as in Experiment 2. The data are visualised in Figure 2 (center). The resulting AUROC/AUPR values are in Table II, and the entries of matrix A estimated with the distribution-based method are visualised in Figure 3. This time the distribution-based method outperforms the pseudotime-based method, and it even attains higher AUROC/AUPR scores than in Experiment 2. This result is expected, since the distribution-based method gets more information from the heterogeneity in the distribution, whereas the pseudotime method erroneously tries to fit the heterogeneity by adjusting the pseudotimes.

D. Experiment 4

In the fourth experiment the cell variability was more realistic. The initial distribution was again a mixture of two Gaussian distributions, with the distance between their means doubled compared to experiment 3, and their covariances were 100Γ. In experiment 4a, the amount of data is as in Table I, and these data are visualised in Figure 2 (right). In experiment 4b, the amount of measured cells at each time point was doubled, resulting in 544 measurements in total, at five different times.

With smaller amount of data in experiment 4a, the best results were surprisingly obtained by method *C*, implying that the other methods were unable to obtain meaningful information from the measurement distributions. Method *B* suffered again of multimodality problems in MCMC sampling and the results were gathered from five independent sampling chains. On the other hand, when the number of measurements was increased in experiment 4b, then the distribution-based method was again the best performer.

It should be noted that with linear systems, the mean of population j is propagated by $e^{A(T_{j+1}-T_j)}$ to the mean of population $j+1$. This is not true with nonlinear systems, and therefore a method tracking the averages of the measured batches is likely to perform worse with nonlinear systems.

IV. Conclusions

Two different paradigms were introduced for identifying linear systems from snapshot ensemble observation data. The first paradigm is based on tracking the evolution of the distributions of cells accross time. The second paradigm is based on the pseudotime concept, where the idea is based on the fact that the cells evolve with different rates and therefore one snapshot contains information from different development stages of the cell. The developed pseudotime-method samples trajectories from which the measurements are obtained at different (pseudo)times (since the cells develop with different rates). On average, the pseudotime-method gave slightly better results than the distribution-based method, and when the data contained only moderate noise (and no model class mismatch), its performance was excellent. However, the distribution-based method seemed to be more robust against disturbances. The pseudotime-method tries to fit the trajectory and the pseudotimes into the data. If the data contain some systemic heterogeneity which

TABLE II: AUROC/AUPR values for the methods *A*: the distribution-based method, *B*: the pseudotime method, and *C*: the batch average method. Each method has some sparsity-enforcing parameter, and the results were established with different values of these parameters. Note that higher λ promotes sparser solutions, whereas higher ρ promotes less sparse solutions.

Experiment		1				2	3	4a	4b
Parameter	λ	0.0025	0.005	0.01	0.05	0.005	0.005	0.005	0.005
Method	*A*	0.923/0.881	0.931/0.900	**0.984**/**0.941**	0.931/0.888	0.751/0.589	0.927/0.868	0.661/0.332	0.913/0.837
Parameter	ρ	0.3	0.25	0.2	0.15	0.2	0.2	0.2	0.2
Method	*B*	**1.000**/**1.000**	0.977/0.899	0.965/0.879	0.981/0.909	0.862/0.639	0.845/0.622	0.733/0.389	0.895/0.753
Method	*C*	0.852/0.696	**0.864**/**0.716**	0.856/0.712	0.842/0.696	0.723/0.310	0.762/0.469	0.761/0.470	0.745/0.404

is not due to the varying developmental rates (such as in Experiment 3), then the method will try to explain the heterogeneity by the pseudotimes, causing an error in the method. Some pseudotime estimation methods are able do detect branches in the biological processes [4]. In such approach, one trajectory only takes into account data belonging to one branch, thus avoiding overfitting. Obviously, the relative performances of the methods may still vary depending on the quality of the data. One observation is that when the cell variability is high (Experiment 4), then sufficiently many measurements are needed in order to obtain information from the distribution of cells. On the other hand, single-cell experimental techniques are developing fast, and the number of measured cells in most experiments far exceeds what we used in the numerical experiments.

Future work includes implementation of nonlinear dynamics either using a mechanistic approach [10] or nonparametric dynamics functions [6], development of a more efficient optimisation scheme for solving (3), and experiments using real data.

REFERENCES

[1] E. Shapiro, T. Biezuner, and S. Linnarsson, "Single-cell techniques will revolutionize whole-organism science," *Nature Reviews Genetics*, vol. 14, no. 9, pp. 618–630, 2013.

[2] C. Trapnell, D. Cacchiarelli, J. Grimsby, P. Pokharel, S. Li, M. Morse, N. Lennon, K. Livak, T. Mikkelsen, and J. Rinn, "The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells," *Nature Biotechnology*, vol. 32, no. 4, pp. 381–391, 2014.

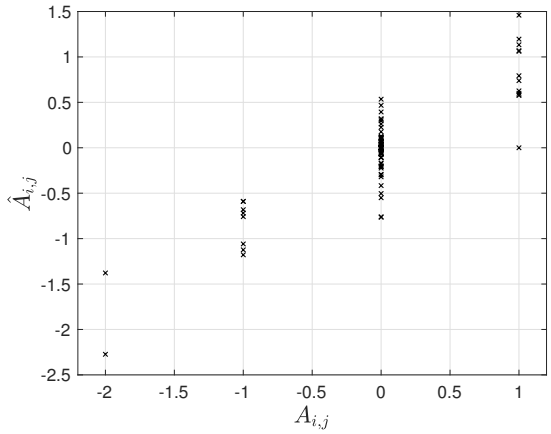

Fig. 3: The elements of A estimated with the distribution-based method in experiment 3 plotted against the true values.

[3] J. Reid and L. Wernisch, "Pseudotime estimation: deconfounding single cell time series," *Bioinformatics*, vol. 32, no. 19, pp. 2973–2980, 2016.

[4] A. Boukouvalas, J. Hensman, and M. Rattray, "BGP: Branched Gaussian processes for identifying gene-specific branching dynamics in single cell data," *Genome Biology*, vol. 19, no. 1, p. 65, 2018.

[5] A. Aalto and J. Gonçalves, "Bayesian variable selection in linear dynamical systems," *ArXiv:1802.05753*, 2018.

[6] A. Aalto, L. Viitasaari, P. Ilmonen, and J. Gonçalves, "Continuous time Gaussian process dynamical models in gene regulatory network inference," *ArXiv:1808.08161*, 2018.

[7] M. Fiers, L. Minnoye, S. Aibar, C. González-Blas, K. Atak, and S. Aerts, "Mapping gene regulatory networks from single-cell omics data," *Briefings in Functional Genomics*, vol. 17, no. 4, pp. 246–254, 2018.

[8] A. Babtie, T. Chan, and M. Stumpf, "Learning regulatory models for cell development from single cell transciptomic data," *Current Opinion in Systems Biology*, vol. 5, pp. 72–81, 2017.

[9] S. Aibar, C. B. González-Blas, T. Moerman, V. A. Huynh-Thu, H. Imrichova, G. Hulselmans, F. Rambow, J.-C. Marine, P. Geurts, J. Aerts, J. van den Oord, Z. K. Atak, J. Wouters, and S. Aerts, "SCENIC: single-cell regulatory network inference and clustering," *Nature Methods*, vol. 14, no. 11, pp. 1083–1086, 2017.

[10] U. Herbach, A. Bonnaffoux, T. Espinasse, and O. Gandrillon, "Inferring gene regulatory networks from single-cell data: a mechanistic approach," *BMC Systems Biology*, vol. 12, no. 1, p. 105, 2017.

[11] S. Zeng, S. Waldherr, C. Ebenbauer, and F. Allgöwer, "Ensemble observability of linear systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 6, pp. 1452–1465, 2016.

[12] A. Küper, R. Dürr, and S. Waldherr, "Dynamic density estimation in heterogeneous cell populations," *IEEE Control Systems Letters*, vol. 3, no. 2, pp. 242–247, 2019.

[13] J. Hasenauer, S. Waldherr, M. Doszczak, N. Radde, P. Scheurich, and F. Allgöwer, "Identification of models of heterogeneous cell populations from population snapshot data," *BMC Bioinformatics*, vol. 12, no. 125, 2011.

[14] G. Schiebinger, J. Shu, M. Tabaka, B. Cleary, V. Subramanian, A. Solomon, J. Gould, S. Liu, S. Lin, P. Berube, L. Lee, J. Chen, J. Brumbaugh, P. Rigollet, K. Hochedlinger, R. Jaenisch, A. Regev, and E. Lander, "Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming," *Cell*, vol. 176, no. 4, pp. 928–943, 2019.

[15] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991.

[16] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society, Ser. B*, vol. 58, pp. 267–288, 1996.