Article

Patterns

Inferring gene regulatory networks from single-cell RNA-seq temporal snapshot data requires higher-order moments

Highlights

- Single-cell RNA-seq temporal snapshot data for detecting regulation
- Challenges in data pre-processing, feature extraction, and network inference for GRNs
- Encoding of regulatory information in higher-order raw moments
- Non-linear least-squares inference for temporal scRNA-seq snapshot data

Authors

N. Alexia Raharinirina, Felix Peppert, Max von Kleist, Christof Schütte, Vikram Sunkara

Correspondence

raharinirina@zib.de (N.A.R.), sunkara@zib.de (V.S.)

In brief

The focus of the paper is to highlight how to decompose a gene regulatory network inference problem into three subproblems (pre-processing, feature extraction, and inference), so that the gene regulatory information is preserved in each step. This way of thinking is key for biological explainability and reproducibility of the inferred gene regulatory networks for data.

Raharinirina et al., 2021, Patterns 2, 100332
September 10, 2021
https://doi.org/10.1016/j.patter.2021.100332

CellPress

Patterns

CellPress
OPEN ACCESS

Article

Inferring gene regulatory networks from single-cell RNA-seq temporal snapshot data requires higher-order moments

N. Alexia Raharinirina,[1,*] Felix Peppert,[2] Max von Kleist,[3] Christof Schütte,[1,4] and Vikram Sunkara[1,2,5,*]
[1]Mathematics of Complex Systems, Zuse Institute Berlin, 14195 Berlin, Germany
[2]Explainable A.I. for Biology, Zuse Institute Berlin, 14195 Berlin, Germany
[3]MF1 Bioinformatics, Methods Development and Research Infrastructure, Robert Koch Institute, 13353 Berlin, Germany
[4]Department of Mathematics and Computer Science, Freie Universität Berlin, 14195 Berlin, Germany
[5]Lead contact
*Correspondence: raharinirina@zib.de (N.A.R.), sunkara@zib.de (V.S.)
https://doi.org/10.1016/j.patter.2021.100332

THE BIGGER PICTURE Single-cell RNA sequencing (scRNA-seq) has become ubiquitous in biology. Recently, there has been a push for using scRNA-seq snapshot data to infer the underlying gene regulatory networks (GRNs) steering cellular function. A recent benchmark of 12 GRN methods demonstrated that the algorithms struggled to predict the ground-truth GRNs and speculated that the low performance was due to the insufficient resolution in the scRNA-seq data. Rather than proposing another method, this paper focuses on how to decompose a GRN problem into three subproblems (pre-processing, feature extraction, and inference), so that the gene regulatory information is preserved in each step. Subsequently, we discuss how to best approach each of the three subproblems.

Concept: Basic principles of a new data science output observed and reported

SUMMARY

Single-cell RNA sequencing (scRNA-seq) has become ubiquitous in biology. Recently, there has been a push for using scRNA-seq snapshot data to infer the underlying gene regulatory networks (GRNs) steering cellular function. To date, this aspiration remains unrealized due to technical and computational challenges. In this work we focus on the latter, which is under-represented in the literature. We took a systemic approach by subdividing the GRN inference into three fundamental components: data pre-processing, feature extraction, and inference. We observed that the regulatory signature is captured in the statistical moments of scRNA-seq data and requires computationally intensive minimization solvers to extract it. Furthermore, current data pre-processing might not conserve these statistical moments. Although our moment-based approach is a didactic tool for understanding the different compartments of GRN inference, this line of thinking—finding computationally feasible multi-dimensional statistics of data—is imperative for designing GRN inference methods.

INTRODUCTION

The emergence of single-cell RNA sequencing (scRNA-seq) technology, the extraction of the transcriptome of individual cells, has helped immensely in detecting and delineating heterogeneities in cells.[1–7] Furthermore, with advances in machine learning and mRNA metabolic tagging, scRNA-seq has provided new insights into cellular development and disease pathogenesis.[1,4–6,8–10] In light of these advances, the development of methods which infer the underlying gene regulatory network (GRN)–which drives cellular decisions—is lagging behind.[11]

Cellular function is dependent on the cell's transcriptomic signature, where the proteins translated from the mRNA form signaling pathways, which perform the cellular function and then, in a feedback loop, regulate the mRNA transcription to translate proteins.[12–14] The process of a gene affecting the expression of another gene is referred to as gene regulation,

and the collection of all gene regulatory interactions (e.g., in a cell) forms a GRN.[1,2,12,15–18] Unlike protein-protein interactions, where educts are converted into products, gene regulatory interactions are more illusive. A gene regulates another gene through its downstream protein complexes, which affect the rate of transcription of the gene being regulated. That is, gene regulation physically occurs on the DNA level and its effect is observed on the mRNA level. In particular, ''gene A regulates gene B'' means that gene A either upregulates (promotes) or downregulates (inhibits) the rate of transcription of gene B. The fact that scRNA-seq only captures mRNA, while gene regulation interactions take place up- or downstream of the mRNA, constitutes a major hurdle for inferring GRNs from scRNA-seq data.[19,20] Recently, a trend has emerged to use multiple temporal scRNA-seq snapshots to capture the underlying GRNs of cells.

Current single-cell GRN inference methods using temporal snapshot data can be grouped into two families, the distribution-based methods and the moment-based methods, distinguished by the type of summary statistics that they use. Distribution-based methods construct their summary statistic based on the empirical distribution of the gene expression in the snapshots,[21–25] whereas moment-based methods utilize only the moments (mean and covariance) of the snapshot data.[26–28]

The rapid increase in the number of GRN inference methods has motivated the development of comprehensive comparative frameworks. A recent benchmark of 12 GRN methods demonstrated that the algorithms struggled to predict the ground-truth GRNs and speculated that the low performance was due to the insufficient resolution in the scRNA-seq data.[11] Rather than proposing another method, the focus of this paper is to dissect and identify key computational stumbling blocks for inferring GRNs from scRNA-seq data.

In general, there are three key components to a GRN inference method: the data pre-processing, the feature extraction, and the inference of the underlying gene regulation pattern. Data pre-processing transforms the data to make it more tractable for analysis, for example, pseudo-time reordering[29] and variational autoencoders.[30] Feature extraction compiles summary statistics of the data, which is intended to contain the information of the regulation, for example the mean or mutual information. Lastly, the inference of the underlying gene regulation pattern is what finds the GRN among the space of all possible GRNs that best matches the statistics of the data, for example, linear least squares or random forest. We take a bottom-up approach and shed light on key challenges in these three core components through intuitive tailored GRN models.

In this work, we construct three different *in silico* GRNs (known ground truth), containing different gene regulation patterns and modeled by Markov-jump processes according to the standard dogma, with the regulation placed upstream from the mRNA.[13,31] Our GRN models consider the intrinsic noise arising from the stochastic nature of gene regulation.[32–35] We designed the models to be nested with ascending order in the number of regulatory reactions, from no regulatory interactions (only correlated through time) to many regulatory interactions (double feedback loop). We then use simulations to produce artificial/synthetic scRNA-seq data on different levels of resolution (e.g., time lags). We omit extrinsic noise in order to investigate, in the best-case scenario, the possibility of reconstructing the three GRN models by four inference methods. Our strategy is based on the reasoning that if a GRN inference method is not able to infer the ground truth for clean artificial data, then it will not be successful for real-world data that contain a multitude of caveats. For comparison, we chose two distribution-based inference methods, mutual information (MI)[22] and SINCERITIES (SINgle CEll Regularized Inference using TIme-stamped Expression profileS),[24] and two moment-based methods, moments derivative-based (linear moment-based inference[27]) and higher-order moment-based (non-linear moment-based inference [this paper and Cao and Grima[36]]).We chose the MI and SINCERITIES methods because they have been proven to be two of the most robust methods in the recent benchmarking by Pratapa et al.;[11] furthermore, our data satisfied the assumptions of these methods. We implemented the linear moment-based inference (linear MBI) method according to the foundation paper by Klimovskaia et al.,[27] which is the best-performing moment-based inference method for GRNs to date. The construction of the non-linear MBI method in this work is an adaptation of the linear MBI method, whereby the derivatives are integrated rather than estimated. The terms ''linear'' and ''non-linear'' refer to the fact that the residual function is linear or non-linear with respect to the parameters being inferred, respectively.

We finish this work by discussing the need for higher-order moments for GRN inference and their computational challenges. We also discuss the caveats of existing datasets and the need for multi-omics data for truly validating GRNs inferred from scRNA-seq temporal snapshot data.

RESULTS

The variance time course is only moderately explained by the mean time course in single-cell datasets

Most GRN inference methods use the mean time course to infer their GRNs. This is historically motivated by bulk RNA-seq, where only the mean transcription levels are observed. However, with single-cell transcriptomics, more complex statistical time courses can be constructed. To illustrate the information captured by more complex statistics, we studied the trends in the mean and variance of gene expression of individual genes from single-cell datasets. We used the variance as a simple surrogate of a non-linear statistics and quantified the correspondence between trends in the mean and the trends in the variance across five different single-cell datasets (see ''trends in the mean and variance of gene expression datasets'' in experimental procedures).

Looking at the human embryonic stem cell (hESC) scRNA-seq dataset, we noticed that the trends of the mean time course and the trends of the variance time course followed similar behavior (Figure 1A). However, when we looked at the correspondence of individual genes, we found that genes that had a similar mean trend exhibited different variance trends; for example, genes following mean trend number 4 (a trend of: up, flat, then down) were distributed mostly across trends a, b, and d in the variance (Figure 1B). Similar patterns were observed for the other datasets. We further quantified this correspondence using the measure of proficiency (also known as normalized mutual information) to compare between the datasets (Figure 1C). We found that real datasets had proficiency in the range of 6%–37%, i.e.,

Figure 1. Mean trends and variance trends in single-cell datasets

(A) Trends in the mean and variance of gene expression for hESC scRNA-seq dataset.

(B) Number of genes in the hESC scRNA-seq dataset associated with the mean and variance trends. The rows and columns have been rearranged using hierarchical clustering to show the overlap between the groups.

(C) Percentage of variance trends explained by mean trends across five different single-cell datasets, using boxplots.

6%–37% of the trends in the variance time courses corresponded with the trends of the mean time courses. Furthermore, we found that the synthetic dataset made through GeneNetWeaver (GNW) had very low proficiency (<5%), which could be attributed to the use of white noise in its model. In contrast, the synthetic dataset from the BoolODE method had high proficiency (>50%), which could be attributed to the use of mean-based colored noise in its model.

In summary, the mean and variance time courses of real gene expression data contained mutually exclusive information. This gives motivation to enrich GRN inference through the use of more complex statistics, for example the higher-order moments.

Stochastic two-gene interaction models

To investigate the role of moments in unraveling regulatory reactions from scRNA-seq data, we constructed three simple two-gene GRN models (see "two-gene interaction model" in experimental procedures). Our first GRN model was a simple no-interaction (No-I) two-gene model, where each gene, Gene A and Gene B, can be in one of two discrete states, on or basal (off state), and can switch between these states via a constant propensity. The gene is then transcribed into mRNA at a constant rate depending on the state of the gene. The transcribed mRNA then undergoes translation and the respective protein is synthesized (Figure 2, top). The mRNA and proteins undergo degradation proportional to their respective populations. In the No-I model, the downstream products associated with their respective gene are not correlated across genes (Table 1). Our second GRN model was a mono-directional interaction (Mono-I) model, i.e., it had the same reactions as the No-I model with the exception of an interaction whereby Protein B actively upregulates the switching off of Gene A (Figure 2, middle). In this scenario, Gene A and its downstream products are affected by the regulation of Gene B; however, Gene B is not affected by any downstream products of Gene A (Table 2). Our third GRN model was the bidirectional interaction (Bi-I) model, where Protein A upregulates the switching off of Gene B and vice versa: Protein B upregulates the switching off of Gene A (Figure 2, bottom). In the Bi-I model, all products in the system are correlated (Table 3).

Covariance and skewness can aid in detecting regulatory pathways

The three models were simulated using the stochastic simulation algorithm (SSA) (see "synthetic scRNA-seq data" in experimental procedures). Only the mRNA expression counts from the simulations were extracted for regulatory inference to mimic scRNA-seq data.

In the No-I model, we observed that both the time course of the mean expression of both mRNAs (A and B) increased identically until the time horizon (Figures 3A and S1A). Due to there being no interactions across genes, as expected, the samples at any fixed time point exhibited near zero covariance between the mRNA expression counts (Figures 3A and 3E). In the Mono-I model, we observed that at early time points the mRNAs' mean expression increased similarly, then the mean expression of mRNA A started to plateau while the mean expression of mRNA B continued to rise, and had a time course similar to that of mRNA B in the No-I model (Figures 3B and S1B). We observed in the time course that the mRNAs had a negative covariance between them (Figures 3B and 3E). Lastly, in the Bi-I model, we observed that the mean expression time course of the mRNAs increased identically, as in the No-I model. The expression distribution was found to also have a negative covariance structure; however, upon inspecting the distribution of a snapshot at $T = 60$ min, we saw that the distribution was very symmetric and was shaped like a waning crescent (Figures 3C and 3E).

To understand the origin of the crescent shape, we compared the time course of the skewness of mRNA A and mRNA B in the three GRN models. We found that in both the Bi-I and No-I models, the mRNA A was positively skewed and followed a similar time course. Furthermore, in the Bi-I model, mRNA B was also skewed similarly to mRNA A. Specifically, all downregulated mRNAs in the models exhibited similar skewness (Figures 3F, S1C, and S1D).

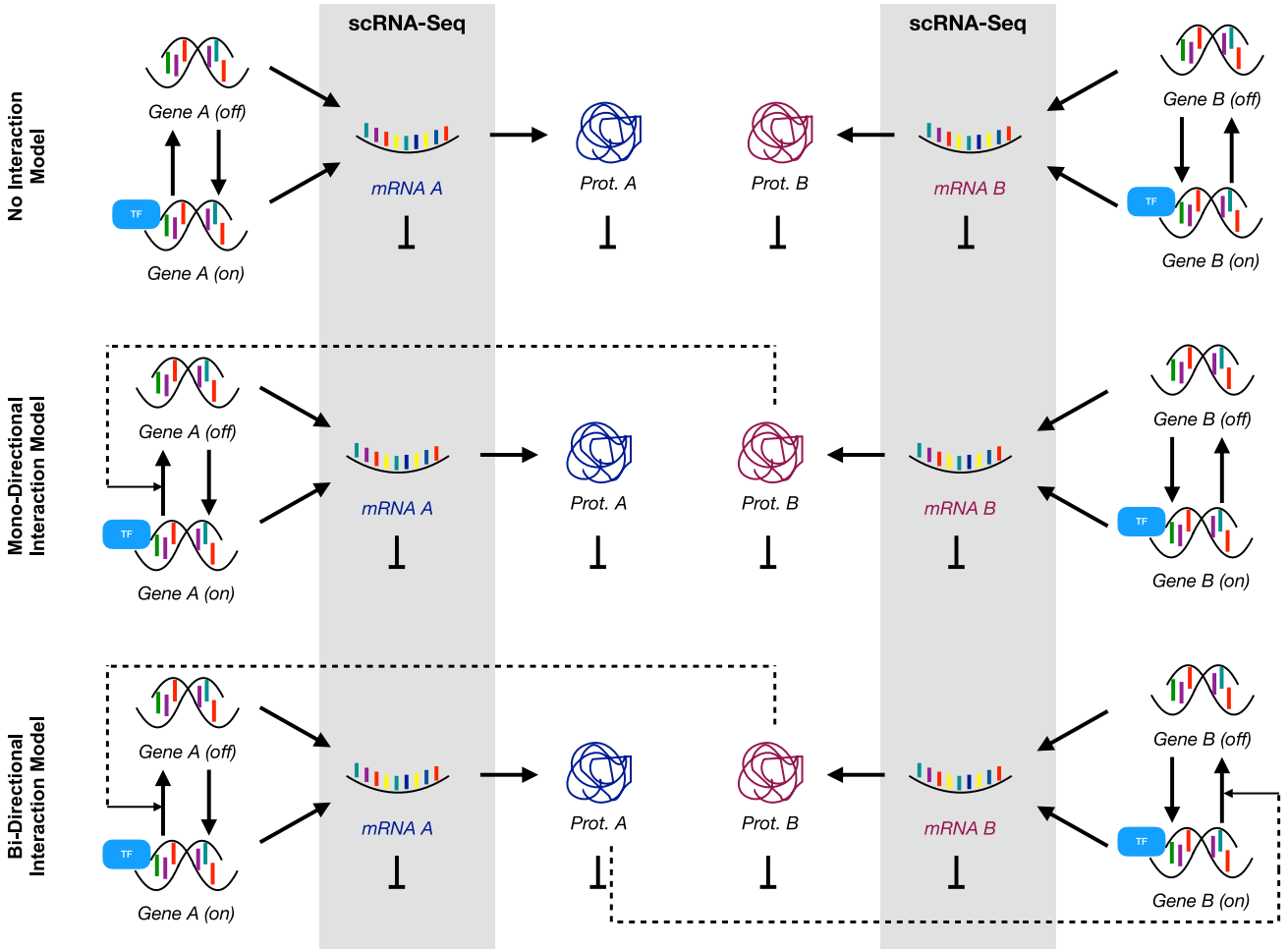

Figure 2. Two-gene interaction models

The three GRN models of interest comprising two genes, A and B, and their corresponding mRNA and proteins. From top to bottom, model schematics are shown for No-I (no interaction), Mono-I (mono-directional interaction), and Bi-I (bidirectional interaction). Only the mRNA counts, shown in the gray shaded area, are used in the GRN inference methods.

In summary, comparing only the mean time course of the three GRN models, we could not distinguish the underlying regulatory reactions between the No-I and Bi-I models. Similarly, the covariance could distinguish that Mono-I and Bi-I had some "negative" interaction occurring relative to the No-I model. However, the direction of the interactions was unclear. When we compared the skewness of the mRNAs, we could see that in the Mono-I interaction mRNA A was being affected, and a similar effect was also acting on both mRNA A and mRNA B in the Bi-I model. Hence, the regulatory information was not in one statistic, but rather distributed over at least three statistics: the mean, the covariance, and the skewness.

Pseudo-time augmented snapshots do not recapitulate the skewness in the original data

When multiple snapshots are unavailable, the temporal information required for GRN inference is typically approximated by pseudo-time augmentation of the expression data. Pseudo-time represents the normalized distance (e.g., diffusion distance—see "pseudo-time ordering of the data" in experimental procedures) from some initial state, or respectively the ordering of the cells along differentiation trajectories. To study whether the pseudo-time augmentation preserves the moments, we removed all true time labels within each of the three GRN models' data and augmented the expression counts with pseudo-time values. In the time course of the central moments of the pseudo-time augmented data, we observed that both the No-I and Bi-I models' mean expression of the mRNAs had a similar trend, as in the true time course (Figures 3G–3I and S1E). With respect to the covariance, we found that pseudo-time augmented data had negative covariance in the Mono-I and Bi-I models. Surprisingly, we found a positive covariance in the No-I model time augmented data (Figures S1F and S1G). The most drastic differences were seen in the skewness, where for all three models the pseudo-time augmented data showed predominantly negative skewness, sharply contrasting with positive skewness seen in the original data. In summary, although pseudo-time augmented data can capture trends in the first two central moments, it could underestimate the skewness, hindering accurate GRN inference.

Table 1. Components of the two-gene no-interaction model

#	Reaction	Coeff. (min^{-1})	Propensity	Stoichiometry	Description
1	Gene A **on** → Gene A **off**	σ_1 = 0.125	σ_1 [Gene A **on**]	(−1,0,0,0,0,0)	inactivation
2	Gene B **on** → Gene B **off**	σ_2 = 0.125	σ_2 [Gene B **on**]	(0,−1,0,0,0,0)	inactivation
3	Gene A **off** → Gene A **on**	σ_3 = 0.5	σ_3 [Gene A **off**]	(1,0,0,0,0,0)	activation
4	Gene B **off** → Gene B **on**	σ_4 = 0.5	σ_4 [Gene B **off**]	(0,1,0,0,0,0)	activation
5	Gene A **on** → Gene A **on** + mRNA A	ρ_1 = 4.75	ρ_1 [Gene A **on**]	(0,0,0,0,1,0)	transcription
6	Gene B **on** → Gene B **on** + mRNA B	ρ_2 = 4.75	ρ_2 [Gene B **on**]	(0,0,0,0,0,1)	transcription
7	Gene A **off** → Gene A **off** + mRNA A	ρ_3 = 1.0	ρ_3 [Gene A **off**]	(0,0,0,0,1,0)	transcription
8	Gene B **off** → Gene B **off** + mRNA B	ρ_4 = 1.0	ρ_4 [Gene B **off**]	(0,0,0,0,0,1)	transcription
9	mRNA A → mRNA A + Protein A	γ_1 = 5.0	γ_1 [mRNA A]	(0,0,1,0,0,0)	translation
10	mRNA B → mRNA B + Protein B	γ_2 = 5.0	γ_2 [mRNA B]	(0,0,0,1,0,0)	translation
11	Protein A → ∅	κ_1 = 0.1	κ_1 [Protein A]	(0,0,−1,0,0,0)	degradation
12	Protein B → ∅	κ_2 = 0.1	κ_2 [Protein B]	(0,0,0,−1,0,0)	degradation
13	mRNA A → ∅	δ_1 = 0.1	δ_1 [mRNA A]	(0,0,0,0,−1,0)	degradation
14	mRNA B → ∅	δ_2 = 0.1	δ_2 [mRNA B]	(0,0,0,0,0,−1)	degradation

The positions in the stoichiometry vector correspond to (Gene A, Gene B, Protein A, Protein B, mRNA A, mRNA B).

Table 2. Modified reactions for mono-directional interaction model

#	Reaction	Coeff. (min^{-1})	Propensity	Stoichiometry	Description
1a	Gene A **on** + Prot. B → Gene A **off**	σ_1 = 0.01875	σ_1 [Prot. B] [Gene A **on**]	(−1,0,0,−1,0,0)	inactivation
5a	Gene A **on** → Gene A **on** + mRNA A	ρ_1 = 6.0	ρ_1 [Gene A **on**]	(0,0,0,0,1,0)	transcription

Reactions to be replaced in Table 1 to obtain the mono-directional interaction model. The positions in the stoichiometry vector corresponds to (Gene A, Gene B, Prot. A = Protein A, Prot. B = Protein B, mRNA A, mRNA B).

No- and mono-directional interactions are more difficult to infer than bidirectional interactions

Four inference methods were applied to infer GRNs from the synthetic scRNA-seq data of the three interaction models: the MI method, the SINCERITIES method, the linear MBI method, and the non-linear MBI method (see "GRN inference methods" in experimental procedures). The numerical experiments were repeated 400 times for robustness and statistical analysis.

The MI method inferred non-zero MI scores for all three models (see "MI method" in experimental procedures). We observed a more than 5-fold increase in the mean edge score for the Bi-I model with respect to the No-I model and, furthermore, the mean edge score for the Mono-I model was found in between (Figure S3A). A one-way ANOVA analysis showed that the differences in the means of the edge scores of the three models were statistically significant (an F value of 49,418 and a p value of strictly less than 0.001). Furthermore, a pairwise comparison with Tukey's HSD (with a p value of 0.001) also showed a significant difference between each pair of models. Using the mean MI score of the No-I model as the minimum score edge cutoff, we concluded that the MI-based approach was effective in detecting that the three models had a different magnitude of interactions between the genes (Figure 4A).

In SINCERITIES, the interaction strength score is estimated by regularized regression of a system of distributional distances, while the sign of interaction (activation versus repression) is determined by the sign of the partial correlation coefficient (see "SINCERITIES method" in experimental procedures). In the No-I data, the SINCERITIES method inferred all possible activations between and within genes with a weak consistency in interaction scores (Figures 4B [left] and S3B). The interaction scores for the Mono-I model gave a clearer result, where a true positive self-activation of mRNA A and mRNA B were observed; however, the repression of mRNA A by mRNA B was missing. Instead, SINCERITIES inferred a false-positive interaction of mRNA A

Table 3. Modified reactions for bidirectional interaction model

#	Reaction	Coeff. (min^{-1})	Propensity	Stoichiometry	Description
1a	Gene A **on** + Prot. B → Gene A **off**	σ_1 = 0.01875	σ_1 [Prot. B] [Gene A **on**]	(−1,0,0,−1,0,0)	inactivation
2a	Gene B **on** + Prot. A → Gene B **off**	σ_2 = 0.01875	σ_2 [Prot. B] [Gene B **on**]	(0,−1,−1,0,0,0)	inactivation
5a	Gene A **on** → Gene A **on** + mRNA A	ρ_1 = 6.0	ρ_1 [Gene A **on**]	(0,0,0,0,1,0)	transcription
6a	Gene B **on** → Gene B **on** + mRNA B	ρ_2 = 6.0	ρ_2 [Gene B **on**]	(0,0,0,0,0,1)	transcription

Reactions to be replaced in Table 1 to obtain the bidirectional interaction model. The positions in the stoichiometry vector correspond to (Gene A, Gene B, Prot. A = Protein A, Prot. B = Protein B, mRNA A, mRNA B).

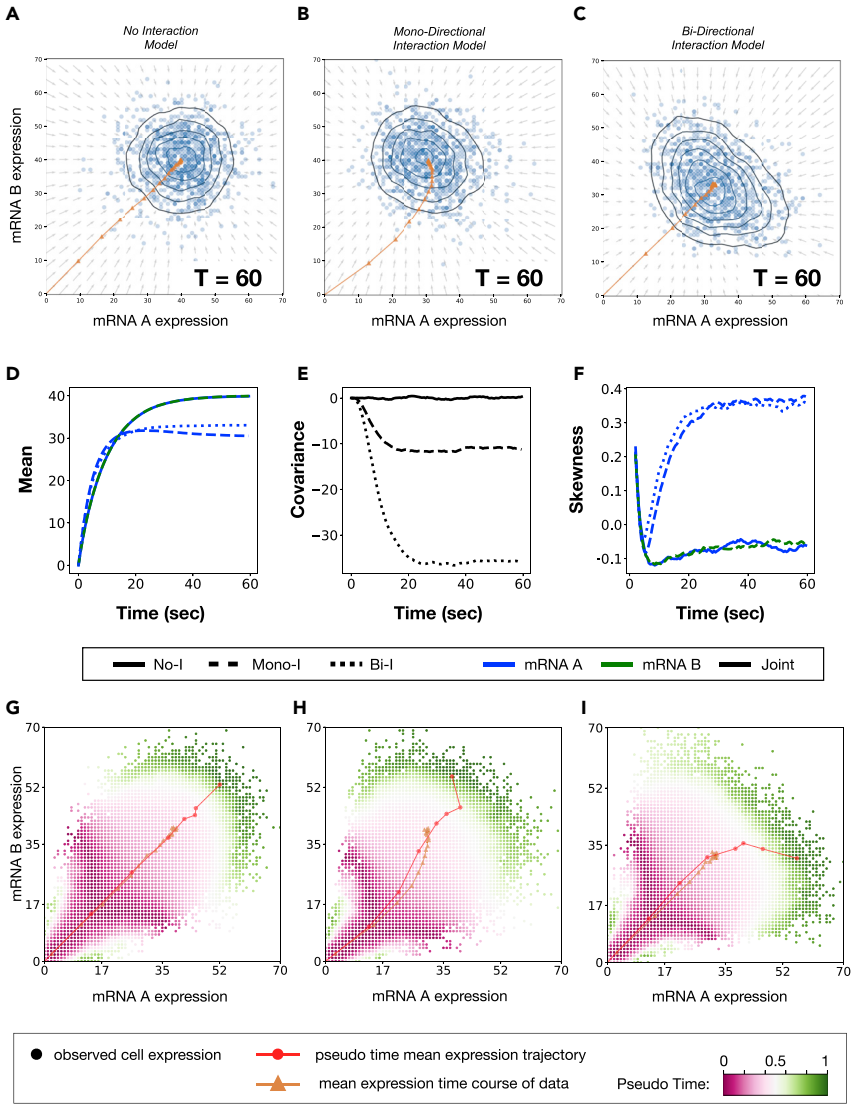

Figure 3. Snapshots of the three interaction models

Snapshot gene expression data at time $T = 60$ showing 1,000 sample mRNA population counts.

(A–C) No-I model (A), Mono-I model (B), and Bi-I model (C). Arrows represent the vector field of the derivative of the first-order moment, and the orange line is the mean expression time course from an initial expression value (mRNA A, mRNA B) = (0, 0) to the time horizon $T = 60$.

(D–F) Mean (D), covariance (E), and the skewness (F) time courses of the three interaction models.

(G–I) Pseudo-time reordering of the data corresponding to (A), (B), and (C), with the red line representing the pseudo-time mean.

See also Figure S1.

activated by mRNA B (Figure 4B, middle). The inference of the Bi-I model was done correctly by SINCERITIES with high interaction scores (Figures 4B [right] and S3C).

The MBI methods used the time course of up to degree three moments to infer their GRNs (see "linear MBI method" and "non-linear MBI method" in experimental procedures). Given we knew a priori that the information was in the first three moments, to avoid overfitting we used 15 times fewer snapshots in MBI than in MI and SINCERITIES.

We observed that linear MBI performed slightly worse than SINCERITIES in inferring the underlying GRNs of the three models (Figure 4C). In particular, the linear MBI method predicted a false-positive activation between mRNA B and mRNA A in the Bi-I model. Looking at individual GRNs inferred in the 400 replicates, we found that the linear MBI method at best inferred the correct GRN 2.5 % of the time (Figure S2B).

Lastly, the non-linear MBI method performed the best out of the four methods. It predicted all true-positive interactions and no false-positive interactions (Figure 4D). Furthermore, looking to the individual GRNs in the replicates, we found that it correctly predicted the No-I model 54 % of the time, the Mono-I model 78 % of the time, and the Bi-I model 95 % of the time (Figure S2C).

To discount that the results were only due to influence of the model parameters, we also performed experiments in which we considered different scenarios with varying regulatory reaction kinetic rates as well as different starting mRNA populations. We found that there were some specific scenarios whereby the inference methods performed better than in the original setting, although overall most methods performed relatively similarly or more poorly than in the original setting (Figures S4 and S5; Notes S1 and S2).

In summary, of the four inference methods that we compared, the Bi-I model was the easiest to capture (Figure 3C). We suspect that this results from the strong double correlation signal present in the data, which results from the non-linear interaction between Gene A and Gene B. The fact that the Mono-I model only had one interaction was detected by all methods; however, the directionality and regulatory mechanism could not be correctly detected. Lastly, the No-I model showed that not all methods are specific enough to correctly detect no interaction.

Linear MBI methods are sensitive to interval lengths between snapshots

The linear least-squares method is well established and can be used to solve high-dimensional inference problems. To harness its scalability for inferring GRNs using moments (linear MBI), good approximations of the derivatives of the moments' time courses are essential (Figure 5A). However, due to snapshot intervals generally being large in sequencing experiments, good derivative approximations are seldom possible. We investigated the effect of interval lengths between snapshots on the accuracy of the inference by constructing a simple stochastic damped oscillator model (see "evaluation of MBI method accuracy" in experimental procedures) (Figures 5B and 5C). Snapshots of

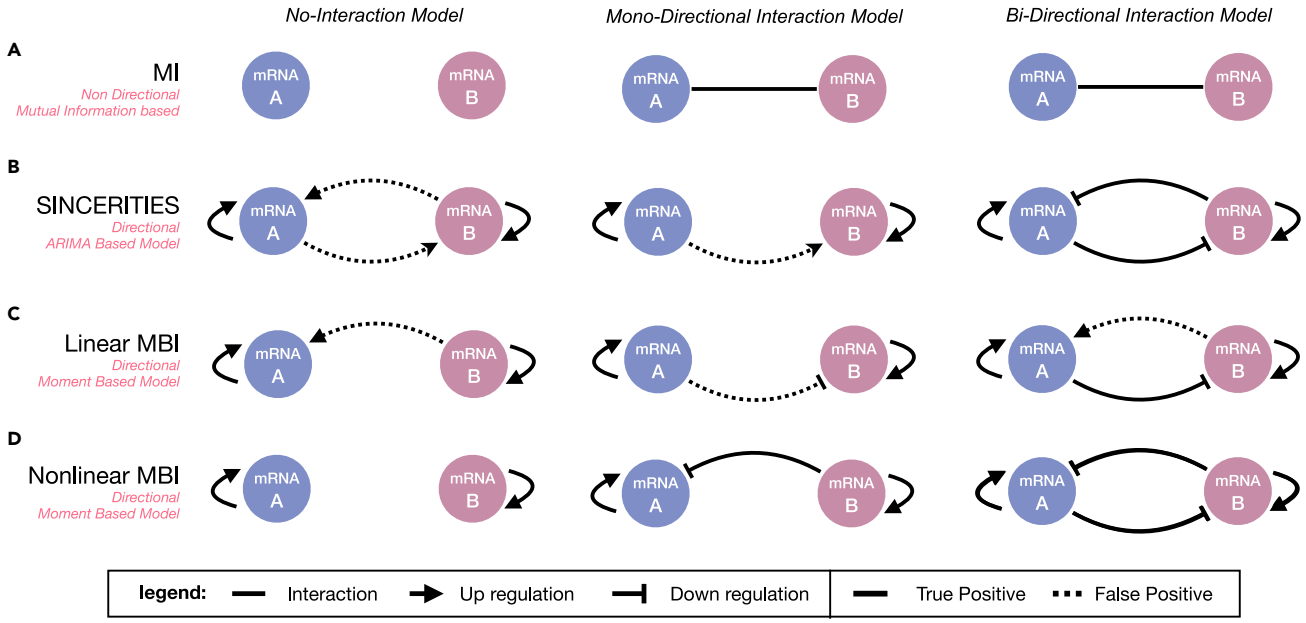

Figure 4. GRNs obtained from four inference methods

The inferred networks corresponding to the data from the three models No-I, Mono-I, and Bi-I are shown in columns from left to right, respectively. The rows correspond to the inference method used: (A) MI method, (B) SINCERTIES, (C) linear MBI, and (D) non-linear MBI. See also Figures S2–S5.

different time-interval lengths were taken and their underlying network was inferred using the linear and non-linear MBI methods.

We observed that the residual sum of squares of the linear MBI method increased with order $\mathcal{O}(h^{0.4})$ with respect to interval length h between snapshots (Figures 5D and 5E). Upon inspecting the inferred reactions, we found that for time interval of $h =$ 0.05 min, the linear MBI method inferred the five true reactions and a further five false-positive reactions (Figure 5H). For the subsequent interval lengths, we found that the linear MBI method continued inferring five to six false-positive reactions and that the number of true-positive reactions was decreasing. The mean time course of the SSA simulations with the inferred parameters showed that the linear MBI method performed poorly in fitting data, even for the smallest interval length of $h =$ 0.05 min (Figure S6). In this case, 128 snapshots (1,152 moments) were used to infer 13 reactions and, surprisingly, we did not observe a close reconstruction of the real data. This suggests that the errors made in estimating the derivative could not be remedied by the large amount of snapshot data.

The non-linear MBI method circumvents the derivative estimation step at the cost of a significant increase in computational time

The non-linear MBI method circumvents the derivative estimation by minimizing the distance of the inferred model to the data. This results in a non-linear least-squares problem, which does not need the time-course derivative of the moments. In comparison with the linear MBI method, for time intervals less than $h = 0.6$ min we found that the non-linear MBI method had at least one order of magnitude lower residual sum of squares in all moments (Figures 5D and 5E). Furthermore, we found that the residual sum of squares did not increase linearly for small time-interval lengths, showing a near flat trend between residual and interval length. Looking at the inferred reaction network, we observed that the non-linear MBI method captured all of the true reactions and only inferred one false-positive reaction for time intervals less than $h = 0.6$ min (Figure 5H). Interestingly, we observed that for intervals larger than $h = 0.8$ min, the non-linear MBI method starts to perform as poorly as the linear MBI method. Upon closer inspection, we found that $h = 0.8$ min is roughly where the first peak in the time course of population A occurs (Figures 5B and 5C). Comparing the Akaike information criterion (AIC) scores of the two approaches, we observed that the non-linear MBI's minimum AIC score was at least two orders of magnitude smaller than that of the linear MBI (Figure 5F). Even though the non-linear MBI method performed better, it must be noted that it took on average nearly 2,000 times longer to compute than the linear MBI method (Figure 5G).

In summary, the simple stochastic damped oscillator model highlighted the major challenges of using moments-based methods for inference. In particular, we observed that the log of the residual scaled sublinearly with the interval length for the linear MBI method. To achieve a similar accuracy as the non-linear MBI method at interval length $h = 0.05$ min, the interval length of the data for the linear MBI would have to be smaller than 10^{-3} min. Furthermore, we found that the snapshot interval has to be small enough to observe the turning points of the system for accurate inference.

DISCUSSION

We considered three GRN models that contained a varying mixture of correlation and regulation between the mRNA populations in time. The highest degree of correlation was contained in the No-I model, a medium mixture of correlation and regulation

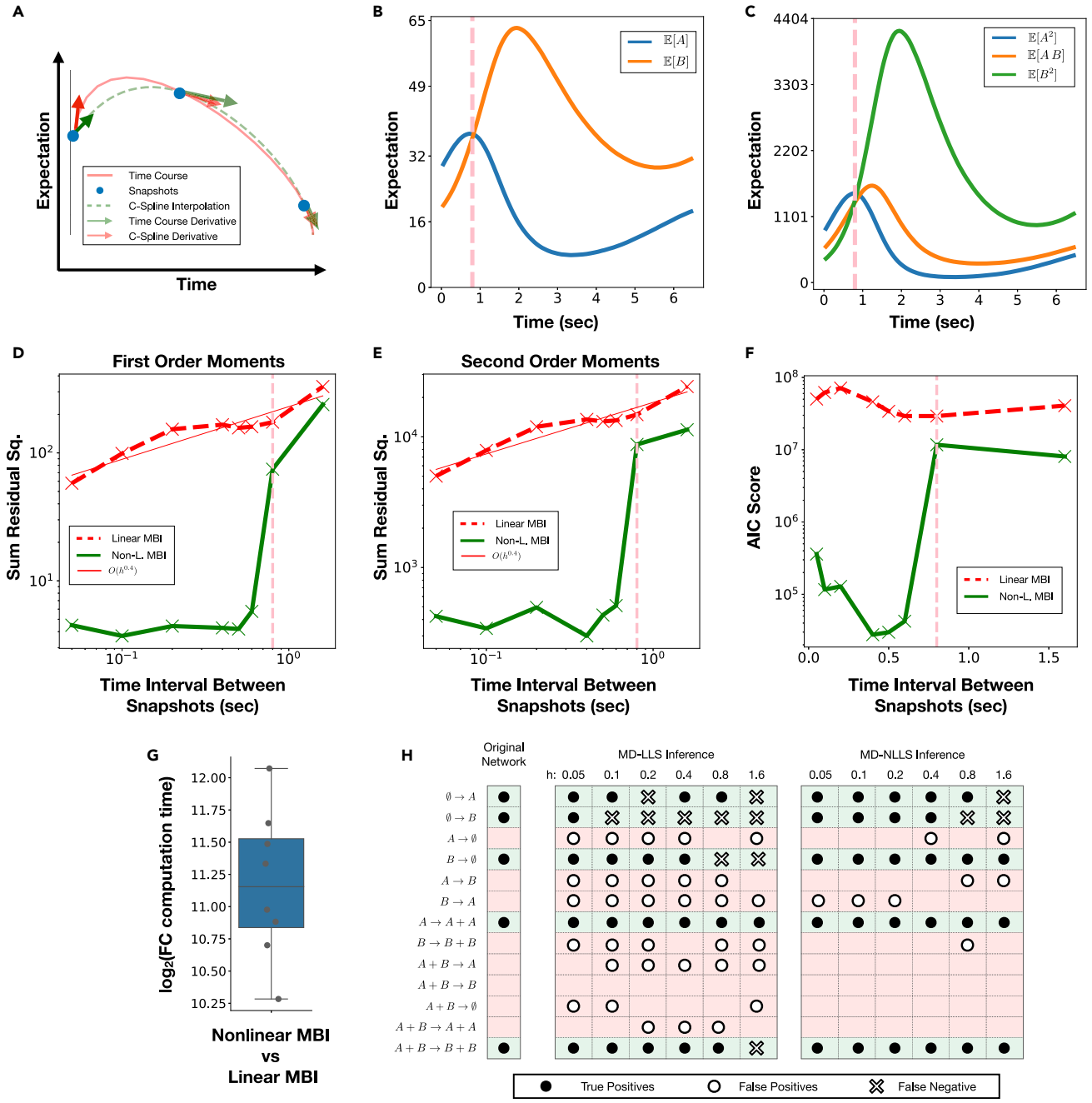

Figure 5. Performance of MBI methods

Performance of linear MBI and non-linear MBI methods on the stochastic damped oscillator model.

(A) Cartoon showing the interpolation and derivative estimation of snapshot time courses.

(B and C) Time course of the first-order (B) and second-order (C) moments. The first peak in the mean time course of population A is denoted by the pink dashed vertical line.

(D and E) Residual sum of squares of the first-order (D) and second-order (E) moments for varying snapshot interval lengths. The linear fit of the residual least squares of the linear MBI method is denoted by the thin red line. The first peak in the mean time course of population A is denoted by the pink dashed vertical line.

(F) AIC score for varying snapshot interval lengths. Linear MBI (red dashed line) and non-linear MBI (green solid line) are shown. The first peak in the mean time course of population A is denoted by the pink dashed vertical line.

(G) Boxplot of the Log_2 fold change in computation time: non-linear MBI versus linear MBI.

(H) True positive (solid dots), false positive (hollow dots), and the false negative (hollow cross) reactions inferred by the linear and the non-linear MBI methods for varying snapshot interval lengths of the stochastic oscillator model.

See also Figure S6.

was contained in the Mono-I model, and the highest degree of regulation was contained in the Bi-I model. We can draw three key conclusions from the experiments conducted in this paper: first, at least up to third-order moments are required for GRN inference; second, only the non-linear MBI method is able to consistently infer the ground-truth GRN of our artificial scRNA-seq data, thereby having high potential to infer GRN of real-world data; third, certain data pre-processing steps such as pseudo-time are not recommended. These three key conclusions are now discussed in more detail.

We observed that regulatory reactions in our models generated distinctive signatures in the higher-order (second- and third-order) moments of the dataset. These signatures, however, could not fully be detected by the distribution-based inference methods (MI and SINCERITIES), because the summary statistics that they utilized were not related to the higher-order moments, which led to underperformance of these inference methods. The MI method is based on threshold strategies which could detect interacting genes; however, it could not infer the direction of the interaction, and smaller interactions are likely to be false negatives due to the threshold cutoff. Similarly, the SINCERITIES method is also based on threshold strategies with the difference that it infers the direction of the regulation. However, the summary statistics used by SINCERITIES falsely detected the correlation as regulation in the No-I and Mono-I models, suggesting that it was not able to distinguish the nuances between correlation and regulation. The linear MBI method performed poorly despite using the moments' derivatives as a summary statistic. We speculate that its underperformance is a consequence of the time interval between the data snapshots not being ideal for the derivative estimation. In contrast to the other methods, the non-linear MBI method was able to consistently reconstruct our three GRN models.

Using higher-order moments has three key impacts on the inference of GRNs from scRNA-seq data. First, higher-order moments can clearly distinguish between correlation and regulation, which is due to regulatory information being present across the higher-order joint moments, unlike correlation. As a consequence, fewer false-positive and false-negative interactions are inferred, leading to more true-positive regulatory reactions. Second, a summary statistic that captures regulation should be based on the moments—for example, if it satisfies Taylor's theorem, it can be arbitrarily well approximated by a polynomial of the moments. Third, to use higher-order moments, the synthetic data model has to be redesigned to be a Markov-jump process, incorporating Poisson intrinsic noise and moving away from mean-driven Gaussian noise.

The higher-order moments proved to be robust summary statistics for regulatory interactions, although they might not be conserved during data pre-processing. Additionally, the fact that they evolve through time non-linearly causes a major computational challenge for the inference. Thus, new numerical schemes to solve high-dimensional non-linear least-squares problems are essential in furthering the field of inferring GRNs.

In regard to limitations, our synthetic models were designed to be ideal with respect to extrinsic noise, to strongly focus on detecting the regulatory signatures, and to study how to extract this signature. However, in practice, real biological data have far more caveats that need to be captured by the synthetic data. Similarly to the synthetic data generator SERGIO,[37] we envisage adapting our Markov-jump model to simulate more realistic synthetic data, including technical issues such as dropouts, Hill's function propensities, multi-gene interactions, and cell cycles. Another limitation of this work is that there was very little focus given to computation. Recently there has been significant methods development in the subfield of parameter inference of GRNs, inferring parameters of fixed GRN models. Given the GRN inference problem is a generalization of the GRN parameter inference problem, there is much scope for translating some of the techniques from the parameter inference of fixed GRN domain across to the GRN inference problem to help mediate computational complexity, for example the introduction of moment closures,[36,38,39] more robust distribution-based summary statistics,[40,41] and Monte Carlo solvers instead of least squares.[42]

A major obstacle for finding good summary statistics (feature extraction) has been the lack of ground-truth GRNs for scRNA-seq datasets.[11,43] True GRNs that were generated from bulk RNA-seq experiments have not been reproduced with single-cell experiments.[44] It is current practice to use protein interaction networks to check for false-positive interaction in the GRNs inferred from scRNA-seq data.[45] Since protein interaction networks only indicate whether a gene is being expressed or several genes are being co-expressed, they are good at filtering genes that are correlated; however, the underlying regulation and directionality is not captured. Additionally, chromatin immunoprecipitation sequencing datasets provide potential binding sites of transcription factors; however, given that a transcription factor has multiple binding sites and can also form complexes, we can establish the directionality of regulation but lose specificity. In conclusion, we advocate that ground-truth datasets require multi-omics single-cell datasets.[43] Such datasets are imperative for calibrating current approaches and designing new GRN inference methods based on single-cell technology.

Limitations of the study

The main aim of the paper is to highlight how to decompose a GRN problem into three subproblems (pre-processing, feature extraction, and inference) so that information is preserved in each step, which encapsulates regulation. The methods we presented have been shown to work on real data by the respective authors, with the exception of non-linear MBI, which is currently computationally infeasible to solve for dimensions greater than a few. In this work, we have generated synthetic data from a very simplified regulatory model; hence, our observed performances are solely descriptive of our simple regulatory model.

EXPERIMENTAL PROCEDURES

Resource availability

Lead contact

Vikram Sunkara. Email: sunkara@zib.de.

Materials availability

Not applicable.

Data and code availability

The data generated and algorithms implemented in this work can be found at https://github.com/vikramsunkara/ScRNAseqMoments.

Trends in the mean and variance of gene expression datasets

We considered three different single-cell snapshot datasets generated from laboratory experiments. These datasets were: (1) a scRNA-seq dataset from hESCs from which we removed genes with more than 50% dropout cells to obtain a dataset of 8,917 genes and 6 snapshot measurements;[46] (2) a qRT-PCR dataset from THP-1 cells containing 48 genes and 8 snapshots;[47] and (3) a qRT-PCR dataset from mouse embryonic stem cells containing 500 genes and 7 snapshots.[48]

We also considered two synthetic datasets used for benchmarking GRN inference methods. One dataset was generated by GeneNetWeaver (GNW).[49,50] It contained 1,000 time series with 21 snapshots of *Escherichia coli* gene expression of 500 genes. The other dataset was obtained using the BoolODE model.[11] We selected the simulation of "linear long" GRN topology, which consisted of 18 genes. This dataset included ten replicate simulations, each containing gene expression measurements for 5,000 cells and the time at which they were sampled. We pooled the replicate simulations to obtain 1,499 snapshots of the gene expression of 50,000 cells.

For each dataset, the time course of the mean and variance of gene expression was computed. We then normalized the time courses of the mean and variance of gene expression to values in the interval $(-1, 1)$ by shifting each time-point value by the first value in the time course, then dividing the shifted values by the maximum magnitude of gene expression of each gene. We further performed principal component analysis on the normalized time courses and projected the datasets onto the lowest number of principal components, explaining around 95% of the total variation.

The projected datasets were then clustered into trends using the *k*-means clustering algorithm in which the number of clusters was chosen according to a discrete uniform distribution between 4 and 8. Finally, we estimated the marginal and joint probability distribution of a gene being associated to a trend in the mean and being associated to a trend in the variance. These probabilities were used to compute the proficiency measure representing the proportion of trends in the variance that can be explained by the trends in the mean. We repeated this procedure 1,000 times to obtain a proficiency distribution for each of the datasets.

Two-gene interaction model

The interaction networks in the three models No-I, Mono-I, and Bi-I were designed to have a nested property, i.e., the No-I model was a subnetwork of the Mono-I model and the Mono-I model was a subnetwork of the Bi-I model. At any time, a gene has a binary state space {**on**, **off**}. The mRNA and proteins are described by their counts; therefore, they have a positive integer state space. The interaction model considered two genes, Gene A and Gene B, and the species that were involved in the reactions are Gene A, Gene B, Protein A, Protein B, mRNA A, and mRNA B.

Table 1 describes each component of the No-I model. The Mono-I interaction model contains all reactions in Table 1 with the changed reaction given in Table 2. In particular, Reaction 1 is replaced with Reaction 1a to involve one of the protein products (here Protein B) which actively upregulates the switching off of Gene A, and Reaction 5 is modified to Reaction 5a. The Bi-I interaction model also contains all reactions in Table 1, with the changes shown in Table 3, i.e., Reaction 1 is replaced with Reaction 1a as in the Mono-I interaction model, and Reaction 2 is replaced with Reaction 2a to include the upregulation of the switching off of Gene B by Gene A, and the propensity coefficients of Reactions 5 and 6 are modified to those of Reaction 5a and 6a.

Fold change in regulation rates

For the scenarios whereby we experimented with the effect of scaled reaction kinetic rates on the inference methods, we used the same non-regulatory reactions as in the base Mono-I and Bi-I models as given in Tables 2 and 3, respectively. However, for the regulatory reactions, we scaled the kinetic rate by 2^{χ}, where $\chi \in \{-1, 1, 2, 3\}$. Hence, for the Mono-I model, we replaced Reaction 1a to Reaction 1b as shown in Table 4. Similarly, for the Bi-I model, we replaced Reaction 1a and Reaction 2a with Reaction 1b and Reaction 2b, respectively, as shown in Table 5. Lastly, for the uneven bidirectional interaction (Un-Bi-I) model, we modified only Reaction 1a of the Bi-I model to Reaction 1c, where we scaled the kinetic rate by 2^{χ}, as shown in Table 6.

Synthetic scRNA-seq data

For each of the two-gene interaction models No-I, Mono-I, and Bi-I, species counts of (Gene A, Gene B, Protein A, Protein B, mRNA A, mRNA B) were generated using the SSA.[51] The initial population configuration was set to (Gene A **on**, Gene B **on**, 0, 0, 0, 0, 0) to mimic the accessibility of the gene on the chromatin, and the initial simulation time was set to 0 min. Gene states and populations counts in the simulations were sampled in time intervals of 0.5 min and up to a time horizon of 60 min. A sample snapshot of SSA mRNA population counts can be interpreted as scRNA-seq data of an individual cell belonging to a fixed cell population. For each model, we generated a total of 100,000 time trajectories of mRNA species counts, thus simulating temporal snapshots of scRNA-seq data of each individual cell for a population of 100,000 cells. We then constructed 400 replicates of this mRNA count dataset. In each replicate, only for the moment-based methods, we subsampled without replacement 10,000 time trajectories from the 100,000 cells. By subsampling trajectories we construct a smoother time course, which is required for a better estimate of the temporal derivative of the moments, which are in turn required for one of the moment-based methods in this work. The underlying GRNs schematics were inferred according to specific rules suited to each of the GRN inference methods that we applied (see "GRN schematics").

Computing the moments

The moments of the snapshot mRNA population were required for moment-based GRN inference methods (see "GRN inference methods"). Let the population of mRNA A in the dth snapshot of the nth cell be denoted by $a_{d,n}$, similarly for mRNA B, $b_{d,n}$. Then for any pair of non-negative integers (l_1, l_2) such that $l = l_1 + l_2$, the lth order moment of the dth snapshot is given by

$$\mathbb{E}\left[\text{mRNA A}(t_d)^{l_1}\ \text{mRNA B}(t_d)^{l_2}\right] \approx \frac{1}{N_c}\sum_{n=1}^{N_c} a_{d,n}^{l_1}\, b_{d,n}^{l_2}, \qquad \text{(Equation 1)}$$

where N_c is the total number of cells in the snapshot sample.

Pseudo-time ordering of the data

We performed pseudo-time ordering using diffusion maps as described in Haghverdi et al.[29] and implemented in SCANPY.[52] This technique is used to achieve a temporal ordering of non-ordered snapshot observations of RNA-seq gene expression.[9,10,28,53] The diffusion map yielded a non-linear dimension reduction and a denoised representation of the high-dimensional gene expression data. The pseudo-time of a cell was defined as the measured diffusion distance from a given root cell, which was assigned a pseudo-time of zero a priori. The pseudo-time ordered data was suitable for lineage branching inference due to the ability of diffusion maps to recover the mean dynamics of gene expression.[29] Furthermore, the application of the pseudo-time method has been extended to infer GRNs.[26,28]

We removed all true time labels in the synthetic mRNA count data (see "synthetic scRNA-seq data") and created a typical pseudo-time ordering on subsampled snapshots as follows: We subsampled from the 100,000 simulated trajectories representing temporal snapshots of mRNA counts for 100,000 cells. We slightly oversampled cells from earlier time points, as our simulation of gene expression converges to equilibrium in later time points, which would bias the pseudo-time approach. Furthermore, we excluded the first snapshot at time 0 min. We generated three datasets comprising the expression of two genes for 50,000 cells. An arbitrary cell from the first time point was chosen to be the root cell of the diffusion pseudo-time ordering. The pipeline was implemented in SCANPY by computing a sparse nearest-neighbor graph (50 neighbors), the observations were embedded in diffusion map space of three dimensions, and the diffusion pseudo-time for each cell was computed using the first two dimensions.

GRN inference methods

We inferred GRNs from the synthetic scRNA-seq data representing temporal snapshots of mRNA counts of each of the two-gene interaction models described in the two-gene interaction model. Since the mRNA counts are the only information available, we are not aiming to reconstruct the two-gene interaction models. Instead, we aim to capture the regulatory relationships between the genes reflected through the interaction of the mRNAs.

MI method

The MI measure quantifies the amount of information shared between two discrete random variables X and Y and is formulated as

Table 4. Modified reactions for varying mono-directional interaction model scenarios

#	Reaction	Coeff. (min^{-1})	Propensity	Stoichiometry	Description
1b	Gene A **on** + Prot. B → Gene A **off**	$\sigma_1 = 0.01875 \times 2^{\chi}$	σ_1 [Prot. B][Gene A **on**]	$(-1,0,0,-1,0,0)$	inactivation

Reaction to be replaced in Table 2 to obtain the mono-directional interaction model. The positions in the stoichiometry vector correspond to (Gene A, Gene B, Prot. A = Protein A, Prot. B = Protein B, mRNA A, mRNA B). The parameter $\chi \in \{-1,1,2,3\}$ is the scaling used for the experimental scenario.

$$I(X,Y) := \sum_{x\in X}\sum_{y\in Y} p(x,y)\log_2\left(\frac{p(x,y)}{p(x)\,p(y)}\right)\ [bit], \qquad \text{(Equation 2)}$$

where $p(x)$ and $p(y)$ are the probability distributions of X and Y, respectively; and $p(x,y)$ is the joint probability distribution of X and Y.

The MI is a symmetric measure; therefore, it has been used to infer non-directed GRN by using it as a score for the confidence of an edge between the genes.[22] Since MI can take any positive value, there is no general way of interpreting its magnitude. A threshold of top scoring network is usually used to infer the underlying networks.

The mRNA count datasets, generated from our two-gene interaction model, were used in the MI GRN inference method. The MI scores were then computed using the implementation provided by Chan et al.[22] Additionally, it was not possible to choose a number of top scoring networks because there was only one possible edge given by the MI method. Therefore, we developed a different way of interpreting inferring the underlying GRNs (see "score-based schematics: MI and SINCERITIES methods").

SINCERITIES method

SINCERITIES has recently been proposed to infer directed GRNs by using temporal snapshots of gene expression data.[24] The SINCERITIES algorithm implements a regularized regression of a system of Kolmogorov-Smirnov distributional distances and assigns a ranked list of scores $(\alpha_i^{(j)})$ representing the influence of gene j on any other gene i in the dataset. A large score indicates a higher confidence that the corresponding edges exists. Furthermore, SINCERITIES infers the direction of the edges, i.e., the nature of the interaction, from the sign of the partial correlation coefficients between each two genes. Similarly to the MI method described above, there exists no general way of interpreting the magnitude of the $(\alpha_i^{(j)})$ scores. As the authors of SINCERITIES leave it to the user to choose a score threshold for drawing an edge between the genes, we chose a generous score cutoff and inferred the underlying GRNs according to specific rules (see "score-based schematics: MI and SINCERITIES methods").

Linear MBI method

In the linear MBI method, we used an adaptation of the SINDy method (supplemental experimental procedures S3) to infer GRNs from the data.[27] We modeled GRNs as Markov-jump processes (supplemental experimental procedures S1), representing the time evolution of N_s species undergoing N_r chemical reactions. Therefore, we aimed to discover the linear dynamical system that defines the moments of the mRNA species counts involved in the reaction network (Equation 4 in supplemental experimental procedures S1).

We thus considered the state vector $m(t_d) \in \mathbb{R}^{\aleph_0}$ containing the raw moments of the N_s species at time t_d for $d \in \{1,\ldots,D\}$, and $\aleph_0$ denotes the cardinality of the countably infinite set $\mathbb{N}$. We then constructed a library of candidate nonlinear functions $\Psi(\mathbf{M})$ as follows:

$$\Psi(\mathbf{M}) = (\Phi_1(\mathbf{M}),\ldots,\Phi_{N_r}(\mathbf{M})) \in \mathbb{R}^{D\times\aleph_0\times N_r}, \qquad \text{(Equation 3)}$$

where $\mathbf{M} = (m(t_1)\ldots m(t_D))^T \in \mathbb{R}^{D\times\aleph_0}$, and

$$\Phi_j(\mathbf{M}) = (\psi_j(m(t_1)),\ldots,\psi_j(m(t_D)))^T \in \mathbb{R}^{D\times\aleph_0} \qquad \text{(Equation 4)}$$

is the vector of stoichiometric moment functions of reaction j,[27] given as

$$\psi_j(m(t_d)) := a_j\, m(t_d) \in \mathbb{R}^{\aleph_0}, \qquad \text{(Equation 5)}$$

where a_j is the design block corresponding to reaction j in the design matrix $\mathbf{A}$ of the moments equations (Equation 4 in supplemental experimental procedures S1). This allowed us to formulate the following linear system:

$$\dot{\mathbf{M}} := \Psi(\mathbf{M})\theta = \sum_{j=1}^{N_r}\Phi_j(\mathbf{M})\theta_j, \qquad \text{(Equation 6)}$$

where $\dot{\mathbf{M}} = (\dot{m}(t_1),\ldots,\dot{m}(t_D))^T \in \mathbb{R}^{D\times\aleph_0}$ are the moments derivatives evaluated at each fixed time point t_d for $d \in \{1,\ldots,D\}$, and $\theta = (\theta_1,\ldots,\theta_{N_r})^T$ is the vector of propensity coefficients of the N_r reactions that we considered.

A finite dimensional approximation of Equation 6 was obtained by truncation of the moments vector. According to Equation 3 in supplemental experimental procedures S1, up to order $l+1$ moments are required to model the time derivatives of up to order l moments. Thus, we can formulate the truncated system,

$$\dot{\mathbf{M}}_l \approx \Psi(\mathbf{M}_{l+1})\theta := \sum_{j=1}^{N_r}\Phi_j(\mathbf{M}_{l+1})\theta_j, \qquad \text{(Equation 7)}$$

where $\mathbf{M}_l = (m_l(t_1),\ldots,m_l(t_D))^T \in \mathbb{R}^{D\times\mathcal{E}(l)}$, with m_l being the vector of up to order l moments and the number of elements in the vector given by

$$\mathcal{E}(l) := \sum_{k=0}^{l}\frac{(N_s+k-1)!}{k!\,(N_s-1)!}.$$

In the linear MBI method, the propensity parameters are considered as the unknown parameters of the linear system of Equation 7 by directly using the moments data M_{l+1}^{data} instead of M_{l+1} and the numerically estimated moments data derivatives $\dot{\mathbf{M}}_l^{data}$ instead of $\dot{\mathbf{M}}_l$.[27] The method is termed linear MBI due to the fact that it infers the propensity parameters θ from the linear system

$$\dot{\mathbf{M}}_l^{data} \approx \Psi\left(\mathbf{M}_{l+1}^{data}\right)\theta := \sum_{j=1}^{N_r}\Phi_j\left(\mathbf{M}_{l+1}^{data}\right)\theta_j. \qquad \text{(Equation 8)}$$

For the GRN inference, we aimed to discover a sparsely connected network that reflects the minimal set of reactions involved in the network,[27,54] because it has been demonstrated that robust GRNs are parsimonious.[55] Therefore, sparse regression minimization techniques are applied to find the sparse parameter vector $\widehat{\theta}$ satisfying

$$\widehat{\theta} = \underset{\theta\geq 0}{\operatorname{argmin}}\frac{1}{2D}\left\|\dot{\mathbf{M}}_l^{data} - \Psi\left(\mathbf{M}_{l+1}^{data}\right)\theta\right\|_2^2 + \alpha\theta_1. \qquad \text{(Equation 9)}$$

Table 5. Modified reactions for varying directional interaction model scenarios

#	Reaction	Coeff. (min^{-1})	Propensity	Stoichiometry	Description
1b	Gene A **on** + Prot. B → Gene A **off**	$\sigma_1 = 0.01875 \times 2^{\chi}$	σ_1 [Prot. B][Gene A **on**]	$(-1,0,0,-1,0,0)$	inactivation
2b	Gene B **on** + Prot. A → Gene B **off**	$\sigma_2 = 0.01875 \times 2^{\chi}$	σ_2 [Prot. A][Gene B **on**]	$(0,-1,-1,0,0,0)$	inactivation

Reactions to be replaced in Table 3 to obtain the bidirectional interaction model. The positions in the stoichiometry vector correspond to (Gene A, Gene B, Prot. A = Protein A, Prot. B = Protein B, mRNA A, mRNA B). The parameter $\chi \in \{-1,1,2,3\}$ is the scaling used for the experimental scenario.

Table 6. Modified reactions for varying uneven bidirectional interaction model scenarios

#	Reaction	Coeff. (min^{-1})	Propensity	Stoichiometry	Description
1c	Gene A **on** + Prot. B → Gene A **off**	$\sigma_1 = 0.01875 \times 2^{\chi}$	σ_1 [Prot. B][Gene A **on**]	$(-1,0,0,-1,0,0)$	inactivation

Reaction to be replaced in Table 3 to obtain the Un-Bi-Interaction model. The positions in the stoichiometry vector correspond to (Gene A, Gene B, Prot. A = Protein A, Prot. B = Protein B, mRNA A, mRNA B). The parameter $\chi \in \{-1,1,2,3\}$, is the scaling used for the experimental scenario.

This problem was approximated by

$$\widehat{\theta} = \underset{\theta \geq 0}{\operatorname{argmin}} \frac{1}{2D} \|\mathbf{M}_l^{data} - \Psi(\mathbf{M}_{l+1}^{data})\theta\|_2^2, \text{ such that } ||\theta||_1 \leq \beta, \quad \text{(Equation 10)}$$

where β is the upper bound on the sum of the parameters and was set to 1,000 in our calculations. In our simulations, we used up to order four moments of the synthetic mRNA counts, i.e., solving Equation 9 for $l = 3$. In addition, we removed the first 30 data points for a more accurate representation of the moments and further reduced the dataset by 15 times to avoid overfitting. The problem in Equation 10 was solved as a quadratic program.[56] Solutions were obtained using Python, the vector operations were handled with the library numpy, the numerical derivatives $\dot{\mathbf{M}}_l^{data}$ were estimated via the spline difference method using the library scipy.interpolate, and the quadratic program was solved using the built-in solver of the library cvxopt. More details can be found at https://github.com/vikramsunkara/ScRNAseqMoments. The underlying GRNs were then generated according to specific rules (see "flux-based schematics: MBI methods").

Non-linear MBI method

Similarly to the linear MBI method, the non-linear MBI method also inferred the parameters of the reaction kinetics of a Markov-jump process from the moments equation (Equation 1 in supplemental experimental procedures S1). Since the moments equation is an infinite-dimensional system, we handled the truncation problem by introducing the interpolations of the higher-order moments from the data source into a truncated system.

We considered m_l as the vector containing moments up to order l. We then approximated the moments equation with

$$\frac{dm_l(t)}{dt} \approx \frac{d\widehat{m}(t|\theta)}{dt} = \overline{\mathbf{A}}(\theta)\widehat{m}(t|\theta) + \mathbf{B}(\theta)u(t,\theta), \quad \text{(Equation 11)}$$

where $\overline{\mathbf{A}}(\theta) \in \mathbb{R}^{\mathcal{E}(l) \times \mathcal{E}(l)}$ is linear in the parameter θ, and is the block of the design matrix $\mathbf{A}$ that comprises the dependency of the derivatives of order l moments on themselves (see Equation 4 in supplemental experimental procedures S1). $\mathbf{B}$ is a rectangular matrix of dimensions $\mathcal{E}(l) \times (\mathcal{E}(l+1) - \mathcal{E}(l))$, and is the block of the design matrix $\mathbf{A}$ that includes the dependency of the derivatives of order l moments on the order $l+1$ moments, and $u(t,\theta)$ is the vector interpolation of the moments of order $l+1$ from the data. The vector $u(t,\theta)$ is in essence the moment closure for the higher-order moments (see supplemental experimental procedures S4 for formalism and commentary on this closure scheme).

Given D temporal snapshots of moments data $\{\widehat{m}_{data}(t_d)\}$ with $d \in \{1,\ldots,D\}$, the parameter $\widehat{\theta}$ that best represents the data in the model Equation 11 can be obtained via maximum likelihood estimation. If we assume that the errors in the moments model are normally distributed, $\widehat{\theta}$ can be obtained by minimizing the negative log likelihood function, which is proportional to

$$\sum_{d=1}^{D} \left|\left| \widehat{m}(t_d|\theta) - \widehat{m}_{data}(t_d) \right|\right|_2^2. \quad \text{(Equation 12)}$$

The problem is then reduced to the non-linear least-squares minimization problem,

$$\widehat{\theta} := \underset{\theta \geq 0}{\operatorname{argmin}} \sum_{d=1}^{D} \|\widehat{m}(t_d|\theta) - \widehat{m}_{data}(t_d)\|_2^2, \quad \text{(Equation 13)}$$

where $\widehat{m}_{data}$ is the vector containing moments up to order l of the data. Unlike in the linear MBI method, in which the objective function of the minimization problem is linear in the parameters θ (Equation 8), the moments $\widehat{m}$ are non-linear functions of the parameters θ and are obtained from solving Equation 11. For this reason, this method is termed non-linear MBI.

We implemented the non-linear MBI method in Python using SciPy.[57] We used the same moments datasets as in the linear MBI method (see previous section), which included up to the fourth order moments of the synthetic mRNA counts data (see "synthetic scRNA-seq data"). The vector $\widehat{m}$ contained up to order $l = 2$ moments. We computed the splines u of the order $l+1 = 3$ moments by specifying the moments' derivative (Equation 11) at the endpoints of the temporal snapshot data, i.e., at t_0 and t_D (this required the moments of order $l+2 = 4$).

A numerical approximation of $\widehat{m}$ was generated by solving Equation 11 for each two adjacent time-point snapshots and fixed θ, i.e., by solving the initial value problem,

$$\frac{d\varphi(\tau)}{d\tau} = \overline{\mathbf{A}}(\theta)\varphi(\tau) + \mathbf{B}(\theta)u(\tau,\theta), \ \tau \in [t_{d-1}, t_d], \quad \text{(Equation 14)}$$

with the initial condition $\varphi(\tau_0) = \widehat{m}_{data}(t_{d-1})$ for each $d \in \{2,\ldots,D\}$.

The solution of Equation 14 yielded the approximation $\widehat{m}(t_d|\theta) \approx \varphi(\tau = t_d)$. This approximated solution was then used to compute the error function in Equation 12. Due to a large difference between the magnitude of order one (the means) and order two moments in the gene interaction model dataset, we found that the minimization problem was ill-conditioned. To combat this numerically, we multiplied the residuals of the order one moments by a constant weight of 40 to avoid its underestimation. The value of 40 is an upper bound to the order one moments in all GRN scenarios; hence, by multiplying with this number, we computed the numerical error function using similar order of magnitudes in the contributions of both order one and order two moments. In this way, the solver is numerically better conditioned. All remaining terms were given a weight constant of 1. A standard least-squares minimization routine was used to find $\widehat{\theta}$ solving Equation 13, particularly using the built-in minimization solver least_square from scipy.optimize (see https://github.com/vikramsunkara/ScRNAseqMoments). The underlying GRN was then inferred using a set of rules that we define below in "flux-based schematics: MBI methods."

GRN schematics

Score-based schematics: MI and SINCERITIES methods

We conducted an ANOVA analysis of the MI score distribution of the three two-gene interaction models No-I, Mono-I, and Bi-I. We then used the mean MI score of the No-I model as a minimum score edge cutoff for the inferred GRNs.

For the SINCERITIES method, the algorithm generated scores that represent the confidence of an edge between two genes. We then chose a generously low threshold of 0.05 to draw an edge. Additionally, the SINCERITIES method provided the direction of the edge. Therefore, we were able to infer GRNs directly from the output of SINCERITIES. We used the algorithm to infer a GRN for each of the 400 replicates of the mRNA count dataset (see "synthetic scRNA-seq data"). The frequencies of the different resulting interaction networks across the 400 runs are depicted in Figure S2A, and we chose the most frequently inferred network as the underlying GRN (Figure 3).

Flux-based schematics: MBI methods

We set up the linear and non-linear MBI methods to infer the reaction network depicted in Table 7. Reactions 6 and 8 represent the upregulation of mRNA A by mRNA B and mRNA B by mRNA A, respectively, while Reactions 9 and 10 represent the downregulation of mRNA A by mRNA B and vice versa. We then generated the adjacency matrix

$$\begin{pmatrix} \text{Self} - \text{Reg of mRNA A} & \text{B Reg A} \\ \text{A Reg B} & \text{Self} - \text{Reg of mRNA B} \end{pmatrix} = \begin{pmatrix} Av_5/Av_3 & Av_6 - Av_9 \\ Av_8 - Av_{10} & Av_7/Av_4 \end{pmatrix}, \quad \text{(Equation 15)}$$

Table 7. Reaction library for interaction of two mRNA species

#	Reaction	Propensity	Stoichiometry	Description
1	$\varnothing \rightarrow A$	θ_1	$(1, 0)$	synthesis of A
2	$\varnothing \rightarrow B$	θ_2	$(0, 1)$	synthesis of B
3	$A \rightarrow \varnothing$	$\theta_3 A$	$(-1, 0)$	degradation of A
4	$B \rightarrow \varnothing$	$\theta_4 B$	$(0, -1)$	degradation of B
5	$A \rightarrow A + A$	$\theta_5 A$	$(1, 0)$	self-catalysis of A
6	$B \rightarrow A + B$	$\theta_6 B$	$(1, 0)$	synthesis of A by B
7	$B \rightarrow B + B$	$\theta_7 B$	$(0, 1)$	self-catalysis of B
8	$A \rightarrow A + B$	$\theta_8 A$	$(0, 1)$	synthesis of B by A
9	$A + B \rightarrow B$	$\theta_9 A B$	$(-1, 0)$	annihilation of A from encounter with B
10	$A + B \rightarrow A$	$\theta_{10} A B$	$(0, -1)$	annihilation of B from encounter with A

For simplicity we refer to mRNA A as A and mRNA B as B.

Table 8. Stochastic damped oscillator reaction library

#	Reaction	Coefficient	Stoichiometry	Description
1	$\varnothing \rightarrow A$	$\theta_1 = 4.0$	$(1, 0)$	birth of A
2	$\varnothing \rightarrow B$	$\theta_2 = 3.0$	$(0, 1)$	birth of B
3	$A \rightarrow \varnothing$	$\theta_3 = 0.0$	$(-1, 0)$	death of A
4	$B \rightarrow \varnothing$	$\theta_4 = 0.7$	$(0, -1)$	death of B
5	$A \rightarrow B$	$\theta_5 = 0.0$	$(-1, 1)$	transition of A to B
6	$B \rightarrow A$	$\theta_6 = 0.0$	$(1, -1)$	transition of B to A
7	$A \rightarrow A + A$	$\theta_7 = 1.25$	$(1, 0)$	self-production of A
8	$B \rightarrow B + B$	$\theta_8 = 0.0$	$(0, 1)$	self-production of B
9	$A + B \rightarrow A$	$\theta_9 = 0.0$	$(0, -1)$	annihilation of B from encounter with A
10	$A + B \rightarrow B$	$\theta_{10} = 0.0$	$(-1, 0)$	annihilation of A from encounter with B
11	$A + B \rightarrow \varnothing$	$\theta_{11} = 0.0$	$(-1, -1)$	annihilation of A and B from encounter
12	$A + B \rightarrow A + A$	$\theta_{12} = 0.04$	$(1, -1)$	birth of A from encounter with B
13	$A + B \rightarrow B + B$	$\theta_{13} = 0.04$	$(-1, 1)$	birth of B from encounter with A

For simplicity we refer to mRNA A as A and mRNA B as B.

where Av_j are the average number of times reaction j fired within the time window. The diagonal of the adjacency matrix (Equation 15) represents the average number of self-catalysis versus degradation for mRNA A and mRNA B. We only considered a non-zero Self-Reg value if the average self-catalysis exceeded 10 within the time window. Since we only aimed to identify self-regulation, the ratio between the self-catalysis and the degradation reaction is used to obtain a positive number, but subtraction could be used instead of a ratio if we wanted to distinguish between self-upregulation (positive values) or self-downregulation (negative values).

In Equation 15, A Reg B represents upregulation versus downregulation for mRNA B by mRNA A, and vice versa for B Reg A. Thus, the sign of these entries determines the direction of the regulation, i.e., whether it is a net up- or down-regulation. We only considered a non-zero regulation value if the net regulation was higher than ten times within the time window.

Evaluation of MBI method accuracy

We investigated the accuracy of the MBI methods by using the stochastic damped oscillator (SDO) model. The interactions involved in the SDO are presented in Table 8. Using the SSA, we generated synthetic population counts starting from an initial population count of (0, 0). The simulations were performed from an initial time of 0 min to a final time of 6.45 min by taking sample population counts at every time intervals of 0.05 min. The moments data, up to order four, were computed from 10,000 SSA trajectories, which yielded $D = 130$ snapshot data points.

Sensitivity of the MBI methods to interval lengths between snapshot

We investigated the effect of the time-interval separation $\Delta t \in \{0.05, 0.1, 0.2, 0.4, 0.5, 0.6, 0.8, 0.9, 1.6\}$ between snapshots by inferring the reaction network parameters $\widehat{\theta}_{\Delta t}$ from datasets which were subsampled using different Δt. The parameters were inferred by fitting up to order three moments of the data. The parameters found by the linear MBI method were used as initial condition for the non-linear MBI for time intervals below 0.8 min. This method was not computationally feasible for time intervals above 0.8 min, so we used the parameters inferred by the non-linear MBI at time interval 0.4 min as an initial condition for those cases. The error function was then computed for every time interval Δt and order l moment as follows:

$$\mathrm{Err}\left(\widehat{\theta}_{\Delta t}, l\right) := \sum_{d=1}^{D} \left|\left| \mathrm{SDO}^{(l)}_{\widehat{\theta}_{\Delta t}}(t_d) - \mathrm{SDO}^{(l)}_{data}(t_d) \right|\right|_2^2, \qquad \text{(Equation 16)}$$

where $\mathrm{SDO}^{(l)}_{data}$ is the vector containing order $l \in \{1, 2, 3\}$ moments of the data and $\mathrm{SDO}^{(l)}_{\widehat{\theta}_{\Delta t}}$ is the order $l \in \{1, 2, 3\}$ moments vector, computed from 1000 SSA trajectories, which were generated with the parameters $\widehat{\theta}_{\Delta t}$ along the full dataset. We computed the error at all 130 snapshots rather than just on the subsampled snapshots in order to observe how well the approaches estimated the unseen data in between the fitted data.

Comparison of the MBI methods

We compared the network inferred from the linear MBI and the non-linear MBI by using the AIC. The AIC was used to rank inference models by considering a trade-off between goodness of fit and overfitting.[58] To account for the small number of snapshot data points fitted in the MBI methods, which decreased from 130 to only 5 as we increased the time-interval separations between snapshots, we used AIC_c, which corrected the original AIC for small sample sizes.[59] It is defined as

$$\mathrm{AIC}_c\left(\widehat{\theta}_{\Delta t}\right) := 2k - \log\left(\mathcal{L}\left(\widehat{\theta}_{\Delta t}\right)\right) + \frac{2k^2 + 2k}{D_{\Delta t} - k - 1}, \qquad \text{(Equation 17)}$$

where k is the number of inferred parameters, $\mathcal{L}$ is the likelihood function, and $D_{\Delta t} \in \{130, 65, 33, 17, 13, 11, 9, 5\}$ is the number of snapshot data points subsampled respectively corresponding to each Δt.

For the MBI methods, the negative log likelihood function was proportional to the sum of errors in the mean and variance. The AIC_c then reduces to

$$\mathrm{AIC}_c\left(\widehat{\theta}_{\Delta t}\right) = 2k + \frac{1}{2}\sum_{d=1}^{D_{\Delta t}} \mathrm{SDO}_{\widehat{\theta}_{\Delta t}}(t_d|\Delta t) - \mathrm{SDO}_{data}(t_d|\Delta t)_2^2 + D_{\Delta t}\sqrt{2\pi} + \frac{2k^2 + 2k}{D_{\Delta t} - k - 1}, \qquad \text{(Equation 18)}$$

where SDO_{data} is the moment vector containing up to order two moments of the data and $\mathrm{SDO}_{\widehat{\theta}_{\Delta t}}$ is the moment vector containing up to order two moments computed from 1,000 SSA trajectories, which were generated with the parameters $\widehat{\theta}_{\Delta t}$ along the full dataset. The argument $(\tau_d|\Delta t)$ is used to indicate that the SDO vectors only contain the snapshot data points subsampled with Δt.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.patter.2021.100332.

ACKNOWLEDGMENTS

V.S. was supported in part by the Berlin Institute for the Foundations of Learning and Data (grant number 01IS18037H, BMBF) and in part by the

Deutsche Forschungsgemeinschaft Cluster of Excellence MATH+ (EXC-2046/1, project ID 390685689) under Project AA1-1. N.A.R. was also funded by the Deutsche Forschungsgemeinschaft Cluster of Excellence MATH+ under Project IN-B1. M.v.K. received funding from the BMBF, grant number 01KI2016.

AUTHOR CONTRIBUTIONS

N.A.R., F.P., and M.v.K. performed the experiments and analyzed the results. V.S., N.A.R., F.P., and C.S. wrote the manuscript. V.S. and C.S. formulated the problem and designed the experiments. All authors contributed to editing and revising the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests. The funders had no role in the design of the study or the decision to publish.

Received: February 3, 2021
Revised: February 23, 2021
Accepted: July 22, 2021
Published: August 18, 2021

REFERENCES

1. Delgado, F.M., and Gómez-Vela, F. (2019). Computational methods for Gene Regulatory Networks reconstruction and analysis: a review. Artif. Intell. Med. *95*, 133–145.
2. Fiers, M.W.E.J., Minnoye, L., Aibar, S., González-Blas, C.B., Atak, Z.K., and Aerts, S. (2018). Mapping gene regulatory networks from single-cell omics data. Brief. Funct. Genomics *17*, 246–254.
3. Ghanbari, M., Lasserre, J., and Vingron, M. (2018). The Distance Precision Matrix: computing networks from non-linear relationships. Bioinformatics *35*, 1009–1017.
4. Giovanni, I., Ramon, M.-B., and Holger, H. (2019). Single-cell transcriptomics unveils gene regulatory network plasticity. Genome Biol. *20*, 110.
5. Hwang, B., Lee, J.H., and Bang, D. (2018). Single-cell RNA sequencing technologies and bioinformatics pipelines. Exp. Mol. Med. *50*, 1–14.
6. Kolodziejczyk, A.A., Kim, J.K., Svensson, V., Marioni, J.C., and Teichmann, S.A. (2015). The technology and biology of single-cell RNA sequencing. Mol. Cell *58*, 610–620.
7. Stegle, O., Teichmann, S.A., and Marioni, J.C. (2015). Computational and analytical challenges in single-cell transcriptomics. Nat. Rev. Genet. *16*, 133–145.
8. La Manno, G., Soldatov, R., Zeisel, A., Braun, E., Hochgerner, H., Petukhov, V., Lidschreiber, K., Kastriti, M.E., Lönnerberg, P., Furlan, A., et al. (2018). RNA velocity of single cells. Nature *560*, 494–498.
9. Qiu, X., Mao, Q., Tang, Y., Wang, L., Chawla, R., Pliner, H.A., and Trapnell, C. (2017). Reversed graph embedding resolves complex single-cell trajectories. Nat. Methods *14*, 979–982.
10. Trapnell, C., Cacchiarelli, D., Grimsby, J., Pokharel, P., Li, S., Morse, M., Lennon, N.J., Livak, K.J., Mikkelsen, T.S., and Rinn, J.L. (2014). The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. Nat. Biotechnol. *32*, 381–386.
11. Pratapa, A., Jalihal, A.P., Law, J.N., Bharadwaj, A., and Murali, T.M. (2020). Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. Nat. Methods *17*, 147–154.
12. Enze, L., Lang, L., and Lijun, C. (2019). Gene regulatory network review. In Encyclopedia of Bioinformatics and Computational Biology, S. Ranganathan, M. Gribskov, K. Nakai, and C. Schönbach, eds. (Elsevier), pp. 155–164.
13. Holehouse, J., Cao, Z., and Grima, R. (2020). Stochastic modeling of autoregulatory genetic feedback loops: a review and comparative study. Biophys. J. *118*, 1517–1525.
14. Kitano, H. (2002). Systems biology: a brief overview. Science *295*, 1662–1664.
15. Barbuti, R., Gori, R., Milazzo, P., and Nasti, L. (2020). A survey of gene regulatory networks modelling methods: from differential equations, to Boolean and qualitative bioinspired models. J. Membr. Comput. *2*, 207–226.
16. Davidson, E.H., Rast, J.P., Oliveri, P., Ransick, A., Calestani, C., Yuh, C.-H., Minokawa, T., Amore, G., Hinman, V., Arenas-Mena, C., et al. (2002). A provisional regulatory gene network for specification of endomesoderm in the sea urchin embryo. Dev. Biol. *246*, 162–190.
17. Streit, A., Tambalo, M., Chen, J., Grocott, T., Anwar, M., Sosinsky, A., and Stern, C.D. (2013). Experimental approaches for gene regulatory network construction: the chick as a model system. Genesis *51*, 296–310.
18. Zheng, G., and Huang, T. (2018). The reconstruction and analysis of gene regulatory networks. Methods Mol. Biol. *1754*, 137–154.
19. Fortelny, N., Overall, C.M., Pavlidis, P., and Freue, G.V.C. (2017). Can we predict protein from mRNA levels? Nature *547*, E19–E20.
20. Vogel, C., and Marcotte, E.M. (2012). Insights into the regulation of protein abundance from proteomic and transcriptomic analyses. Nat. Rev. Genet. *13*, 227–232.
21. Bonnaffoux, A., Herbach, U., Richard, A., Guillemin, A., Gonin-Giraud, S., Gros, P.-A., and Gandrillon, O. (2019). WASABI: a dynamic iterative framework for gene regulatory network inference. BMC Bioinformatics *20*, 220.
22. Chan, T.E., Stumpf, M.P.H., and Babtie, A.C. (2017). Gene regulatory network inference from single-cell data using multivariate information measures. Cell Syst. *5*, 251–267.e3.
23. Kim, S. (2015). ppcor: an R package for a fast calculation to semi-partial correlation coefficients. Commun. Stat. Appl. Methods *22*, 665–674.
24. Papili Gao, N., Ud-Dean, S.M.M., Gandrillon, O., and Gunawan, R. (2018). SINCERITIES: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. Bioinformatics *34*, 258–266.
25. Spetch, A.T., and Li, J. (2017). LEAP: constructing gene co-expression networks for single-cell RNA-sequencing data using pseudotime ordering. Bioinformatics *33*, 764–766.
26. Aubin-Frankowski, P.-C., and Vert, J.-P. (2020). Gene regulation inference from single-cell RNA-seq data with linear differential equations and velocity inference. Bioinformatics *36*, 4774–4780.
27. Klimovskaia, A., Ganscha, S., and Claassen, M. (2016). Sparse regression based structure learning of stochastic reaction networks from single cell snapshot time series. PLOS Comput. Biol. *12*.
28. Matsumoto, H., Kiryu, H., Furusawa, C., Ko, M.S.H., Ko, S.B.H., Gouda, N., Hayashi, T., and Nikaido, I. (2017). SCODE: an efficient regulatory network inference algorithm from single-cell RNA-Seq during differentiation. Bioinformatics *33*, 2314–2321.
29. Haghverdi, L., Büttner, M., Wolf, F.A., Buettner, F., and Theis, F.J. (2016). Diffusion pseudotime robustly reconstructs lineage branching. Nat. Methods *13*, 845–848.
30. Eraslan, G., Simon, L.M., Mircea, M., Mueller, N.S., and Theis, F.J. (2019). Single-cell RNA-seq denoising using a deep count autoencoder. Nat. Commun. *10*, 390.
31. Cao, Z., and Grima, R. (2020). Analytical distributions for detailed models of stochastic gene expression in eukaryotic cells. Proc. Natl. Acad. Sci. U S A *117*, 4682–4692.
32. Ko, M.S. (1991). A stochastic model for gene induction. J. Theor. Biol. *153*, 181–194.
33. McAdams, H.H., and Arkin, A. (1997). Stochastic mechanisms in gene expression. Proc. Natl. Acad. Sci. U S A *94*, 814–819.
34. Swain, P.S., Elowitz, M.B., and Siggia, E.D. (2002). Intrinsic and extrinsic contributions to stochasticity in gene expression. Proc. Natl. Acad. Sci. U S A *99*, 12795–12800.
35. Thattai, M., and van Oudenaarden, A. (2001). Intrinsic noise in gene regulatory networks. Proc. Natl. Acad. Sci. U S A *98*, 8614–8619.
36. Cao, Z., and Grima, R. (2019). Accuracy of parameter estimation for autoregulatory transcriptional feedback loops from noisy data. J. R. Soc. Interf. *16*, 20180967.

37. Dibaeinia, P., and Sinha, S. (2020). SERGIO: a single-cell expression simulator guided by gene regulatory networks. Cell Syst. *11*, 252–271.e11.

38. Fröhlich, F., Thomas, P., Kazeroonian, A., Theis, F.J., Grima, R., and Hasenauer, J. (2016). Inference for stochastic chemical kinetics using moment equations and system size expansion. PLOS Comput. Biol. *12*.

39. Soltani, M., Vargas-Garcia, C.A., and Singh, A. (2015). Conditional moment closure schemes for studying stochastic dynamics of genetic circuits. IEEE Trans. Biomed. Circuits Syst. *9*, 518–526.

40. Munsky, B., Li, G., Fox, Z.R., Shepherd, D.P., and Neuert, G. (2018). Distribution shapes govern the discovery of predictive models for gene regulation. Proc. Natl. Acad. Sci. U S A *115*, 7533–7538.

41. Öcal, K., Grima, R., and Sanguinetti, G. (2019). Parameter estimation for biochemical reaction networks using Wasserstein distances. J. Phys. A Math. Theor. *53*, 034002.

42. Schmiester, L., Schälte, Y., Fröhlich, F., Hasenauer, J., and Weindl, D. (2019). Efficient parameterization of large-scale dynamic models based on relative measurements. Bioinformatics *36*, 594–602.

43. Lähnemann, D., Köster, J., Szczurek, E., McCarthy, D.J., Hicks, S.C., Robinson, M.D., Vallejos, C.A., Campbell, K.R., Beerenwinkel, N., Mahfouz, A., et al. (2020). Eleven grand challenges in single-cell data science. Genome Biol. *21*, 31.

44. Tanay, A., and Regev, A. (2017). Scaling single-cell genomics from phenomenology to mechanism. Nature *541*, 331–338.

45. Padi, M., and Quackenbush, J. (2015). Integrating transcriptional and protein interaction networks to prioritize condition-specific master regulators. BMC Syst. Biol. *9*, 80.

46. Chu, L.-F., Leng, N., Zhang, J., Hou, Z., Mamott, D., Vereide, D.T., Choi, J., Kendziorski, C., Stewart, R., and Thomson, J.A. (2016). Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. Genome Biol. *17*, 173.

47. Kouno, T., de Hoon, M., Mar, J.C., Tomaru, Y., Mitsuoki, K., Suzuki, H., Hayashizaki, Y., and Shin, J.W. (2013). Temporal dynamics and transcriptional control using single-cell gene expression analysis. Genome Biol. *14*, R118.

48. Stumpf, P.S., Smith, R.C., Lenz, M., Schuppert, A., Müller, F.-J., Babtie, A., Chan, T.E., Stumpf, M.P., Please, C.P., Howison, S.D., et al. (2017). Stem cell differentiation as a non-Markov stochastic process. Cell Syst. *5*, 268–282.e7.

49. Marbach, D., Prill, R.J., Schaffter, T., Mattiussi, C., Floreano, D., and Stolovitzky, G. (2010). Revealing strengths and weaknesses of methods for gene network inference. Proc. Natl. Acad. Sci. U S A *107*, 6286–6291.

50. Marbach, D., Schaffter, T., Mattiussi, C., and Floreano, D. (2009). Generating realistic in silico gene networks for performance assessment of reverse engineering methods. J. Comput. Biol. *16*, 229–239.

51. Gillespie, D.T. (1976). A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J. Comput. Phys. *22*, 403–434.

52. Wolf, F.A., Angerer, P., and Theis, F.J. (2018). SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. *19*, 15.

53. Magwene, P.M., Lizardi, P., and Kim, J. (2003). Reconstructing the temporal ordering of biological samples using microarray data. Bioinformatics *19*, 842–850.

54. Hoffmann, M., Fröhner, C., and Noé, F. (2019). Reactive SINDy: discovering governing reactions from concentration data. J. Chem. Phys. *150*, 025101.

55. Leclerc, R.D. (2008). Survival of the sparsest: robust gene networks are parsimonious. Mol. Syst. Biol. *4*, 213.

56. Gaines, B.R., Kim, J., and Zhou, H. (2018). Algorithms for fitting the constrained lasso. J. Comput. Graph. Stat. *27*, 861–871.

57. Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat. Methods *17*, 261–272.

58. Akaike, H. (1974). A new look at the statistical model identification. IEEE Trans. Automat. Contr. *19*, 716–723.

59. Burnham, K.P., and Anderson, D.R. (2004). Multimodel inference: understanding AIC and BIC in model selection. Sociol. Methods Res. *33*, 261–304.