ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 3 - 2026

Structure Identification and Dynamical Reconstruction of Gene Regulatory Networks Based on Neural Ordinary Differential Equations

Kangsheng Song[1,*], *Qiankun Liu*[1] *and Zhihong Gao*[1]

[1] College of Computer Science and Technology, Shandong University of Technology, Zibo 255000, Shandong, China

SUMMARY: *To solve the split between structural recovery and continuous dynamics' characterisation in gene regulatory network identification, we put forward a Neural ODE-GRN joint modelling method which is based on neural ordinary differential equations. This method uses time-series bulk RNA-seq and pseudo-time-series single-cell RNA-seq expression matrices as input materials, introduces continuous-time evolution equations into a hidden state space, and combines sparse structural masks, symbolic constraints, biological priors and temporal consistency loss to realize co-optimisation of regulatory adjacency matrix learning and expression trajectory reconstruction. Therefore, validation works were conducted with simulated data from GeneNetWeaver and SERGIO, as well as real-world data from hESCs and mESCs. Hence, results show that on a 100-gene simulated network, the model's AUPR achieved 0.631, which is an improvement of 0.042 to 0.114 compared with various baselines; under 20% noise conditions, the AUROC kept above 0.80; on real-world data, the trajectory correlation coefficient reached 0.914-0.928, at the same time maintaining good topological consistency for hub nodes and representative regulatory modules. Therefore, this method provides a new computational tool for explaining dynamic regulatory mechanisms in developmental differentiation and disease progression.*

KEYWORDS: *Neural ordinary differential equations; Gene regulatory network inference; Dynamic reconstruction; Single-cell transcriptomics; Causal regulatory structure*

1 Introduction

Gene regulatory networks (GRNs) are used to characterise the organisational relationships between transcription factors, target genes and the direction of their regulation, and serve as a crucial analytical framework for understanding the mechanisms underlying cellular fate determination, disease progression and stress responses. With the continuous accumulation of single-cell transcriptomic, single-cell multi-omics and time-series expression data, researchers are no longer content with merely reconstructing a 'who-connects-to-whom' network diagram from static expression matrices; instead, they are increasingly focused on the triggering sequences, intensity of action and state transition patterns of regulatory interactions within continuous biological processes. In other words, the questions addressed by GRN identification are expanding from "whether an edge exists" to "how an edge functions, when it takes effect, and how the system evolves over time". In this sense, network structure reconstruction and dynamical description are not two independent tasks, but rather two representations of the same regulatory process at the topological and evolutionary levels.

*19862520287@163.com
https://doi.org/10.65102/is20261131 Published: 10 June, 2026

Advances in single-cell multi-omics research have further demonstrated that real regulatory processes often exhibit characteristics such as sparsity, heterogeneity, non-linearity, and phasic amplification; relying solely on static correlation patterns makes it difficult to fully elucidate the regulatory logic of complex systems [1]. Concurrently, the introduction of multi-modal information has shifted regulatory inference from 'expression covariation' towards 'mechanistic constraints', with the research focus moving from single-network reconstruction to the joint modelling of structure, state, and function [2]. A review of recent methodological developments reveals that GRN reconstruction has moved beyond traditional statistical association frameworks into a new phase integrating deep learning, causal analysis and dynamic systems modelling. However, under the conditions of high noise, low sampling density and continuous state evolution, how to ensure at the same time the structural reliability and dynamic interpretability still is an unsolved challenge inside this field [3].

Existing GRN inference methods have constructed a relatively abundant technical range. Research taking interpretability as the core tries to make model decisions match with biological regulatory evidence, thus making sure that the identification of network edges is not only limited to black-box outputs, hence it is based as much as possible on traceable proofs like feature importance, local rules, or candidate regulatory pathways [4]. Furthermore, Boolean threshold models, mutual information, and their improved algorithms retain some value in the inference of small-to-medium-sized networks, as they possess strong expressive power regarding discrete regulatory logic, non-linear correlations, and local dependency structures [5, 6]. However, from the angle of continuous-time regulation processes, such methods commonly face three restrictions: Firstly, most models depend on likenesses among static samples or discrete state changes to make inference, thus making it hard to precisely show the smooth advance of cellular states along pseudo-time or real-time lines; secondly, the prediction of network edges and the fitting of gene expression paths are usually handled as separated processes, leading to structure-level results lacking dynamical support, and hereby dynamical-level results are difficult to trace back to a clear regulation topology; thirdly, under sparse sampling, dropout noise and asynchronous observation conditions, strengthened correlation does not equal strengthened regulation, and simple edge scoring systems are easy to bring in false correlations. For research situations that need to explain 'when regulation happens, how it accumulates, and finally drives state changes', only static edge identification is no longer enough to satisfy the needs of mechanism analysis.

To address these shortcomings, deep learning methods have gradually become a central focus in GRN research. Sequence representation models, exemplified by Transformers, can uncover cross-gene dependencies from high-dimensional expression patterns and, to some extent, improve edge identification accuracy and local interpretability [7]; graph neural networks further regard genes as graph nodes; by means of message-passing mechanisms, they capture higher-order topological connections, hereby providing a more natural structural representation for the modeling of complex regulatory relationships [8]; Graph autoencoders, meanwhile, attempt to learn network generation mechanisms in a low-dimensional latent space, balancing representation compression with edge recovery capabilities [9]; more lately constructed frameworks based on deep language models regard gene expression patterns as “semantic sequences” that can be modeled, therefore enhancing the models’ capacity to capture long-range dependent relationships and context-related information [10]. These advancements have significantly propelled GRN inference into high-dimensional, non-linear and structurally complex scenarios; however, their common shortcomings are also evident: On the one hand, the majority of models still put their focus on whether directed or undirected edges exist or not, laying strong emphasis on 'network reconstruction' but rarely carrying out direct restriction on 'plausibility of continuous dynamics'; on the other hand, although a lot of

methods can generate satisfactory AUROC or AUPR scores, they do not conduct further verification on whether the obtained networks are capable of supporting trajectory reconstruction, key state replay, or system-level topological feature preservation. Therefore, thus, improvements in performance metrics do not have the inherent implication that regulatory mechanisms are being characterized with higher accuracy. Without constraints on continuous-time evolution, a model may learn a network that is acceptable at the edge level but inconsistent at the dynamic level; without an examination of topological stability and robustness, the model's generalisation ability on real biological data is also unlikely to be convincing.

Based on this understanding, this study introduces Neural Ordinary Differential Equations (Neural ODE) into the GRN identification task, attempting to solve the two core problems of 'structure identification' and 'dynamics reconstruction' simultaneously within a unified framework. Unlike discrete cascaded networks, Neural ODEs can describe the evolution of hidden states in a continuous-time manner, treating gene expression changes not as simple jumps between discrete slices, but as a smooth dynamical process driven by an underlying regulatory field. Building on this, the research further introduces a sparse structural mask to constrain the learnable range of the regulatory network, incorporates prior regulatory constraints to reduce biologically unsupported redundant connections, and integrates adjacent observation times, inter-time-step integration results, and the overall trajectory morphology into a unified optimisation objective via a temporal consistency loss. Consequently, the model no longer learns merely the likelihood at a single time point, but rather a continuous system capable of explaining how the expression state is continuously driven and modulated by the regulatory network. The significance of this approach lies in the fact that the recovered network edges can be verified through dynamical reconstruction, whilst the trajectory fitting results can be traced back to explicit structural parameters; these two aspects mutually constrain one another, thereby enhancing the model's mechanistic consistency and the reliability of its results.

Based on above-mentioned design, the core content of this paper can be summed up into three aspects. First, we build a combined Neural ODE-GRN framework which is customized for continuous-time single-cell data and time-series expression data; through the cooperative action of hidden state evolution, structural sparsity restrictions, and prior masking, we realize overall modeling of regulatory systems. Second, at the structure level, we restore directed regulatory edges and their strength relations, meanwhile at the dynamic level, we rebuild the time-based changing process of gene expression trajectories, thus making unified explanation of the 'network graph' and 'evolutionary curves' inside one single model possible. Third, via experiments using both simulated data and real data, we assess the model’s effectiveness in multiple dimensions – including edge recognition accuracy, trajectory rebuilding error, topological faithfulness, and robustness under noise and sparse sampling – with the purpose of providing an analytical method that balances structural interpretability and dynamic verifiability for decoding regulatory mechanisms in complex biological processes. According to this method, the following parts will successively introduce model construction, joint optimization strategy, experimental arrangement, and results analysis.

2 Methods

2.1 Neural ODE-based continuous-time modeling and structural parameterization

To evade the situation that gene regulatory network inference is reduced to a static edge-ranking task, the put-forward framework implants time-series bulk RNA-seq data or pseudotime-ordered single-cell RNA-seq data into a continuous-time state space, hence it constructs a differentiable regulatory evolution system in the latent domain. Let the observed expression vector be denoted by $x(t)$, where each component corresponds to the normalized expression level of one gene at time t. Since real regulatory processes evolve continuously rather than as isolated jumps between adjacent observations, a latent state vector $z(t)$ is introduced to represent the internal regulatory configuration of the system at time t, and use ordinary differential equations to describe its variation over time, as shown in Eq. (1).

$$\frac{dz(t)}{dt} = f_\theta(z(t), A, u(t)) \tag{1}$$

where $z(t)$ is the latent state vector at time t, dimensionless; t is the time variable, in min; f_θ is the latent-state dynamics function; θ is the parameter set of the dynamics function, dimensionless; A is the regulatory adjacency matrix to be inferred, dimensionless; and $u(t)$ is the external input vector at time t, dimensionless.

The key point of Eq. (1) is that A directly participates in latent-state evolution, rather than being inferred afterward from feature importance scores. Under this formulation, network structure is not an auxiliary output but an integral part of the state flow field. As a result, the questions of whether a regulatory edge exists and how the system state changes can be handled within the same dynamical framework [11, 12]. For real time-series data, $u(t)$ can represent stimulation or environmental perturbation; for pseudotime single-cell data, $u(t)$ can be set to zero so that the model focuses on endogenous regulatory progression.

To ensure that the resulting network is interpretable, this paper does not employ fully free dense connections, but instead parameterises the adjacency matrix with a sparse structure. For any candidate edge, its elements are as shown in Eq. (2).

$$\alpha_{ij} = m_{ij} s_{ij} w_{ij} \tag{2}$$

where α_{ij} is the regulatory coefficient from gene j to gene i, dimensionless; m_{ij} is the prior mask coefficient, taking values of 0 or 1; s_{ij} is the sign constraint coefficient, taking values of +1 or −1; w_{ij} is the learnable edge weight, dimensionless; and i and j are gene indices.

Eq. (2) decomposes each candidate regulatory link into three components: existence permission, regulatory sign, and effect magnitude. This formulation makes it easier to incorporate transcription factor priors, literature-supported evidence, or motif-based candidate relations into the network search space, while preserving the distinction between activating and inhibitory regulation. If an edge violates prior knowledge, m_{ij} =0; if the regulatory sign is known, s_{ij} can be preset; otherwise, the remaining part is learned from data [13]. The resulting A therefore combines sparsity, directionality, and prior-injectable structure,

making it more suitable for the structure identification task.

In terms of the specific implementation, this paper first uses an adjacency matrix to perform structural clustering of the latent states, and then generates state transition rates via a non-linear mapping, as shown in Eq. (3).

$$r(t) = Az(t) \tag{3}$$

where $r(t)$ is the structure-aggregated regulatory input vector at time t, dimensionless.

Therefore, the changing speed of the hidden status is provided by Eq. (4).

$$\frac{dz(t)}{dt} = \sigma(Wr(t) + Bu(t) + b) \tag{4}$$

where $\sigma(\cdot)$ is the nonlinear activation operator; W is the latent-state mapping parameter matrix, dimensionless; B is the external-input mapping parameter matrix, dimensionless; and b is the bias vector, dimensionless.

Equations (3) and (4) jointly indicate that the network structure first determines the regulatory input received by each gene and then governs subsequent continuous state propagation. Therefore, changes in edge weights continuously reshape the trajectory over the whole integration interval, rather than producing only local effects at individual sampling points. This is also the main distinction between the proposed framework and approaches that first fit trajectories and only then attempt to interpret network structure: here, what is learned is a continuous dynamical system under explicit topological constraints, not a generic temporal representation detached from network structure.

To map the hidden state back to the observation space, this paper defines an output mapping function, as shown in Eq. (5).

$$\hat{x}(t) = g\phi(z(t)) \tag{5}$$

where $\hat{x}(t)$ is the reconstructed expression vector at time t, dimensionless; $g\phi(\cdot)$ is the observation mapping function; and ϕ is the parameter set of the observation mapping, dimensionless.

Eq. (5) converts the continuous latent state into an observable quantity that can be aligned with the original expression matrix. It should be emphasized that the goal here is not merely to reconstruct expression trajectories. Instead, expression reconstruction is used as an external test of structural plausibility: only when the topology encoded by A can drive $z(t)$ to produce a continuous evolution consistent with real observations should the inferred structure be regarded as credible. Thus, the core of this section is not trajectory fitting alone, but the direct involvement of the adjacency matrix in ODE evolution, so that network structure inference and dynamical reconstruction are accomplished within one unified framework.

2.2 Joint optimization strategy for network inference and dynamic reconstruction

After we have built up the continuous-time dynamical model, the core problem in training work is not any more to reduce a single fitting error to the smallest degree, but to push network structure recovery and expression trajectory reconstruction to reach convergence under one unified objective. Therefore, we use a mini-batch sequence input-continuous integration-joint backpropagation plan to train the model. First, the encoder generates the

initial latent state; thus, the Neural ODE solver carries out integration of the latent dynamics along the observed time points; finally, the predicted outputs are assessed from four complementary angles: expression reconstruction error, temporal transition inconsistency, structural sparsity, and prior-constraint violation. Therefore, this design can avoid two common failure situations: learning smooth trajectories without interpretable regulatory structure, and getting a sparse network that cannot support realistic dynamic reconstruction. Since atlas-scale external information and multi-omic prior knowledge can effectively narrow the search space of candidate edges, prior masks are directly added into the training objective instead of being used only in post hoc edge screening [14].

The expression reconstruction term is used to ensure that the model output remains consistent with the observed expression at each sampling time; its definition is shown in Eq. (6).

$$L_r = \frac{1}{nK}\sum_{i=1}^{n}\sum_{k=1}^{K}\left\|\hat{x}_i(t_k) - x_i(t_k)\right\|_2^2 \tag{6}$$

where L_r is the expression reconstruction loss; n is the number of samples; K is the number of observed time points; $\hat{x}_i(t_k)$ is the reconstructed expression vector of sample i at time t_k; $x_i(t_k)$ is the corresponding observed expression vector; and t_k is the kth observation time point, in min.

Although Eq. (6) can force the model to fit the expression curves, it does not by itself guarantee that the direction of state propagation between adjacent time points is correct. Consequently, a temporal transition consistency term is introduced to ensure that the latent state resulting from single-step integration remains consistent with the encoded latent state at the next observation point, as shown in Eq. (7).

$$L_d = \frac{1}{n(K-1)}\sum_{i=1}^{n}\sum_{k=1}^{K-1}\left\|\psi_{h_k}\left(z_i(t_k)\right) - z_i(t_{k+1})\right\|_2^2 \tag{7}$$

where L_d is the temporal transition consistency loss; $\psi_{h_k}(z_i(t_k))$ denotes the ODE integration operator with step size h_k; $z_i(t_k)$ is the latent state vector of sample i at time t_k; and $h_k = t_{k+1} - t_k$ is the interval between adjacent observations, in min.

To suppress redundant edges and maintain biological interpretability, the adjacency matrix is further regularised using sparse terms and prior constraint terms, as shown in Eq. (8).

$$\begin{cases} L_s = \|A\|_1 \\ L_p = \|A \odot (1 - \mathrm{M})\|_1 \end{cases} \tag{8}$$

where L_s is the sparsity regularization term; L_p is the biological prior constraint loss; A is the inferred regulatory adjacency matrix; M is the prior mask matrix, whose elements take values of 0 or 1; $\odot$ denotes the Hadamard product; and 1 is the all-ones matrix. Through Eq. (8), the model is encouraged to remove unnecessary links while assigning additional penalties to edges that violate candidate transcription factor relationships or external biological priors. This coupled strategy of representation learning and structural regularization is consistent with recent studies that combine matrix factorization and graph

modeling for network recovery [15].

Therefore, the overall training objective is expressed as shown in Eq. (9).

$$L = \lambda_1 L_r + \lambda_2 L_d + \lambda_3 L_s + \lambda_4 L_p \tag{9}$$

where L is the total loss function, and λ_1, λ_2, λ_3 and λ_4 are the weighting coefficients of the four loss components.

In practice, these coefficients are not kept fixed throughout training. Instead, a two-stage optimization schedule is adopted. During the first 20 epochs, greater emphasis is placed on L_r and L_d so that the latent flow field can first capture the major trend of the observed trajectories. In the subsequent stage, λ_3 and λ_4 are gradually increased to prune redundant edges and strengthen prior consistency. This schedule reduces the risk of underfitting that may arise when structural penalties dominate too early in training. In addition, for the purpose of lifting robustness condition under high dropout rates and irregular sampling, we carry out mild dropout augmentation and Gaussian noise perturbation on the input expression matrix, and multiple starting points will be randomly chosen to conduct integration consistency checks. For networks with strong feedback loops, we do not set a strict acyclicity constraint; therefore, when a more static causal interpretation is wanted, a weak cycle penalty may be introduced hence. Moreover, an upper-bound regularization on the Jacobian spectral radius can be added for suppressing locally explosive dynamics. Recent multiview attention-based studies have shown that relying only on local pairwise dependencies may overlook cross-scale regulatory patterns; accordingly, temporal consistency and prior-guided structural constraints are both retained here to balance local fitting and global structural stability [16]. From the perspective of hyperparameter transfer, meta-learning-based GRN studies also suggest that reusing well-performing initialization settings across datasets can reduce tuning instability and improve training efficiency [17, 18].

To visually illustrate the roles of different loss terms during optimization, Fig. 1 presents the normalized convergence curves of the four loss components. The Fig. is not intended for model comparison, but for showing how the reconstruction term, dynamic consistency term, sparsity term, and prior term cooperate across early, middle, and late training stages.

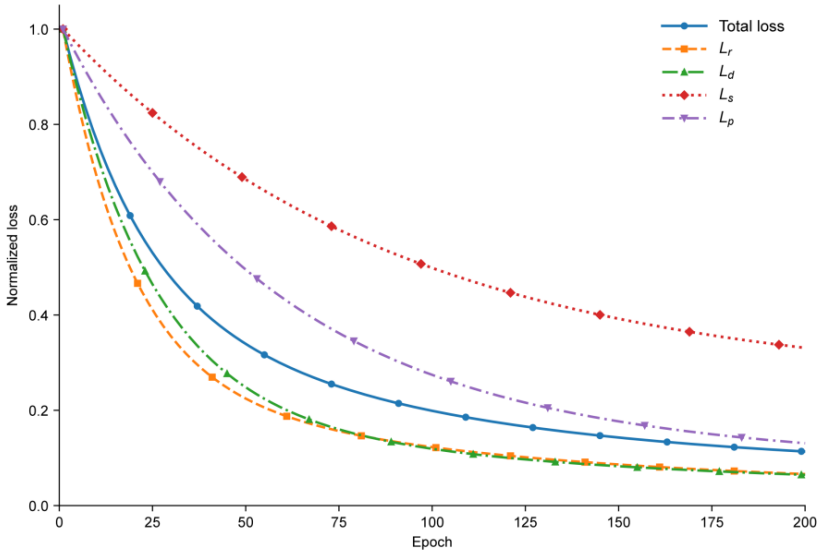

Figure 1: Normalized convergence curves of different loss terms during joint optimization.

As displayed in Fig. 1, the total loss reduces with the fastest speed inside the first 50 epochs, this fact indicates that the model first grasps the main form of the expression trajectories. Therefore, after this period, L_d and L_p keep on falling down, thus suggesting that temporal transition consistency and prior-constrained structure polish become the dominant optimization powers step by step. By comparison, L_s goes down in a slower way, this is in accordance with the common seeing that network sparsification usually falls behind coarse trajectory fitting. Hence, this convergence pattern has very good agreement with the two-stage training strategy introduced above, and it provides the optimization foundation for the later joint assessment of structure recognition and dynamic rebuilding.

2.3 Experimental design, datasets, baselines, and evaluation metrics

To examine correctness of the put forward model in aspects of structural recognition and kinetic rebuilding, both simulated and real data are adopted in the experimental part. Simulated data were produced by GeneNetWeaver and SERGIO so as to supply controllable real networks and continuous observation sequences; therefore, the first is fit for building regulatory topologies with different sizes, thus the second can create simulated data that is more close to single-cell expression profiles under given regulatory network conditions, hence it is often used in GRN inference benchmarking. For real-world data, publicly accessible developmental scRNA-seq datasets were chosen, with hESC (GSE75748) and mESC (GSE98664) being validation objects; these datasets have been widely applied as benchmarks in a lot of single-cell GRN studies. To make consistency with continuous-time modelling easy, pseudo-time series were all rescaled into the 0-100 min interval. To guarantee that data arrangement matches experimental tasks, the composition and purpose of datasets used in this study are expressed clearly in Table 1

Table 1: Experimental Datasets and Task Configurations.

Dataset	Data Source	Data Type	Scale Settings	Time/Trajectory Settings	Main Purpose
GNW-50	GeneNetWeaver	Simulated Regulatory Network	50 genes, 5 network groups	0-100 min, interval 10 min	Small-scale structure recovery
GNW-100	GeneNetWeaver	Simulated Regulatory Network	100 genes, 5 network groups	0-100 min, interval 10 min	Medium-scale structure recovery
GNW-200	GeneNetWeaver	Simulated Regulatory Network	200 genes, 5 network groups	0-100 min, interval 10 min	Large-scale structure recovery
SERGIO-sc	SERGIO	Simulated Single-Cell Expression	Corresponding to the above networks	Continuous trajectory sampling	Dynamics and noise testing
hESC	GSE75748	Real scRNA-seq	High-variance genes and TF candidates after preprocessing	Pseudotime mapped to 0-100 min	Developmental trajectory validation
mESC	GSE98664	Real scRNA-seq	High-variance genes and TF candidates after preprocessing	Pseudotime mapped to 0-100 min	Cross-data generalization validation

In Table 1, the simulated data have the function to display the concept of 'verifiable structure', whereas the real-world data are used to show 'interpretable dynamics'. Therefore, this setting is helpful for discussing edge recovery accuracy, trajectory reconstruction error and topological stability inside one single experimental framework; hence, it also can reduce the conclusion bias that may be caused by depending solely on one single data source.

The comparison includes traditional machine learning, pseudo-temporal ODE models, and latest deep learning methods, that is GENIE3, GRNBoost2, PIDC, SCODE, DeepRIG, STGRNS, scGREAT, scMGATGRN, and GRACE. GRACE, which is introduced in, combines structural causal models with graph neural networks, and hence it is suitable to be a baseline based on causality mechanism; STGRNS, scGREAT and scMGATGRN respectively stand for Transformer, multi-layer context modelling and multi-view graph attention methods, and thus they can well cover main technical branches of current deep learning methods. Therefore, for making the positioning of compared methods convenient, Table 2 gives a summary of categories and key comparison points of each baseline.

Table 2: Comparison of method settings and key points.

Method	Method Category	Key Features	Comparison Focus
GENIE3	Tree Model	Recovery of regulatory edges based on feature importance	Static edge identification baseline
GRNBoost2	Tree Model	Gradient boosting for accelerated inference	Large sample efficiency comparison
PIDC	Information Theory	Utilizes multivariate mutual information	Recovery of nonlinear correlations
SCODE	ODE Method	Based on pseudotime linear dynamics	Continuous time baseline
DeepRIG	Deep Learning	Graph representation and edge prediction	Deep structure recovery
STGRNS	Transformer	Sequence dependency modeling	Long-range dependency learning
scGREAT	Deep Context Model	Enhanced gene context representation	Global representation capability
scMGATGRN	Graph Attention	Multi-view graph modeling	Cross-view structure integration
GRACE	Causal Graph Model	Fusion of causal mechanisms and GNN	Causal interpretability

In Table 2, the chosen baselines contain both classical methods and present main-stream deep learning models; therefore, this enables a comprehensive assessment of the put-forward model in three dimensions: static inference, continuous dynamics, and causal interpretation.

The evaluation metrics are separated into three groups. For the structural layer, AUROC, AUPR, F1, and Early Precision are adopted to measure directed edge recovery ability; the dynamics layer uses RMSE, MAE, Dynamic Time Warping distance and trajectory correlation to judge the consistency between the reconstructed curve and the observed trajectory; the topology layer applies degree distribution error, hub recovery rate, network efficiency deviation and assortativity deviation to describe the faithfulness of the obtained network’s whole structure. Robustness examination was carried out in four aspects: first, through setting dropout rates of 10%, 20% and 30%; secondly, via introducing noise intensities of 0.05, 0.10 and 0.20; thirdly, by exerting non-uniform sampling on the

observation points; and fourthly, through randomly deleting 1 or 2 key time points. All methods were operated on the same set of candidate TFs and under the same data partitioning, thus the mean and standard deviation of five repeated experiments are reported.

3 Results and Discussion

3.1 Network structure identification performance under synthetic benchmarks

On the simulation benchmark, our method is made comparison with GRIT, SLIVER, IGEGRNS, DeepRIG and STGRNS. GRIT, represents a differential Eq. fitting method for dynamic single-cell data [19]; SLIVER is applied to assess large-scale causal structure recovery abilities [20]; IGEGRNS stands for a graph embedding-driven deep learning method [21]. Therefore, given that reporting only edge-level metrics cannot fully reflect network quality, hence the results analysis also includes the hub recovery rate to supplement the assessment of topological fidelity [22]. Fig. 2 shows the comprehensive results for different network scales, noise levels, sampling densities, and hub recovery performance.

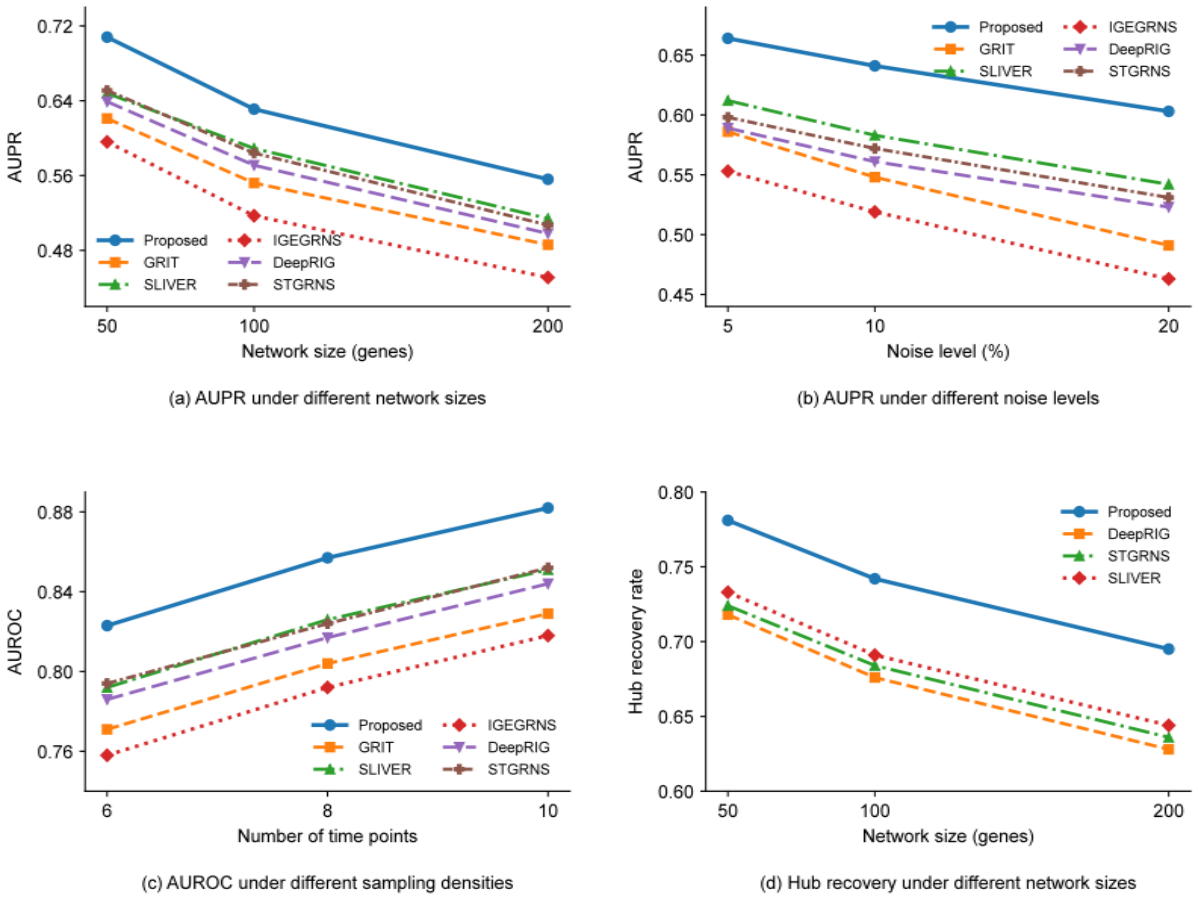

Figure 2: Comparison of network structure recognition performance under simulated conditions.

As displayed in Fig. 2(a), along with network size growing from 50 to 200 genes, all methods have a decline trend, yet the proposed method keeps a consistent superior position. Therefore, its AUROC/AUPR values are 0.901/0.708 for 50 genes, 0.872/0.631 for 100 genes,

and 0.841/0.556 for 200 genes. In the 100-gene situation, the AUPR is higher than that of GRIT, SLIVER, IGEGRNS, DeepRIG, and STGRNS by 0.079, 0.042, 0.114, 0.060, and 0.047 in sequence, hence it suggests that the framework based on Neural ODE maintains strong edge-ranking ability even when the candidate space becomes larger. Fig. 2(b) further demonstrates that the proposed method still owns robustness under noise condition. When noise increases from 5% to 20%, the AUPR drops from 0.664 to 0.603, which is a reduction of 9.2%. By comparison, GRIT decreases from 0.586 to 0.491 and IGEGRNS from 0.553 to 0.463, while at 20% noise, SLIVER, DeepRIG, and STGRNS can only reach 0.542, 0.523, and 0.531. Thus, this more gentle degradation indicates that embedding structural parameters into continuous-time dynamics can enhance resistance to expression perturbation and help keep discrimination ability under noisy environments.

The differences resulting from variations in sampling density are shown in Fig. 2(c). When the number of observation points increased from 6 to 10, the AUROC of our method improved from 0.823 to 0.882, an increase of 0.059; GRIT improved from 0.771 to 0.829, SLIVER from 0.792 to 0.851, DeepRIG from 0.786 to 0.844, and STGRNS from 0.794 to 0.852. It can be seen that a greater number of time points helps various methods improve structural recovery accuracy; however, the method proposed in this paper retains a high baseline even under low sampling density, leading STGRNS by 0.029 and DeepRIG by 0.037 when only 6 time points are available, indicating that it is relatively less dependent on sparse observations.

Fig. 2(d) furthermore makes a comparison on the recovery abilities of hub genes. The hub recovery ratios of our method on networks with 50, 100, and 200 genes are 0.781, 0.742, and 0.695 separately; in opposite, DeepRIG reaches 0.718, 0.676, and 0.628, while STGRNS gets 0.724, 0.684, and 0.636. In the aspect of relative improvement, our method is better than DeepRIG by 8.8%-10.7% and STGRNS by 7.9%-9.3%. Therefore, these results show that the recovered networks not only obtain higher scores at the edge level but also keep the true structure more closely regarding key regulatory nodes. Therefore, a comprehensive analysis of results in the four groups shown in Fig. 2 discovers that the method put forward in this paper shows good stability under three types of stress conditions: network scale expansion, noise enhancement and sampling thinning. Hence, this indicates that continuous-time dynamical constraints play a large role in increasing the reliability of structural identification.

3.2 Dynamic reconstruction accuracy and biological interpretability on real expression data

The evaluation of the results on real-world data focused on three aspects: trajectory reconstruction accuracy, the quality of differentiation branch fitting, and the interpretability of regulatory directions. Based on the pseudo-time series results for the two datasets (hESC and mESC), the average RMSE of the method in this paper was 0.084, the MAE was 0.061, and the trajectory correlation was 0.928 for hESC; and 0.091, 0.067 and 0.914, respectively, for mESCs. Compared with the baseline method, GRLGRN achieved average Pearson r values of 0.887 and 0.874 across the two datasets, representing improvements of 0.041 and 0.040, respectively; In high-dropout regions, the local fluctuation fit is smoother, which is consistent with the observation that dropout augmentation can enhance the robustness of single-cell GRN inference [23]. Fig. 3 presents a comparison of the true and predicted trajectories for four representative genes.

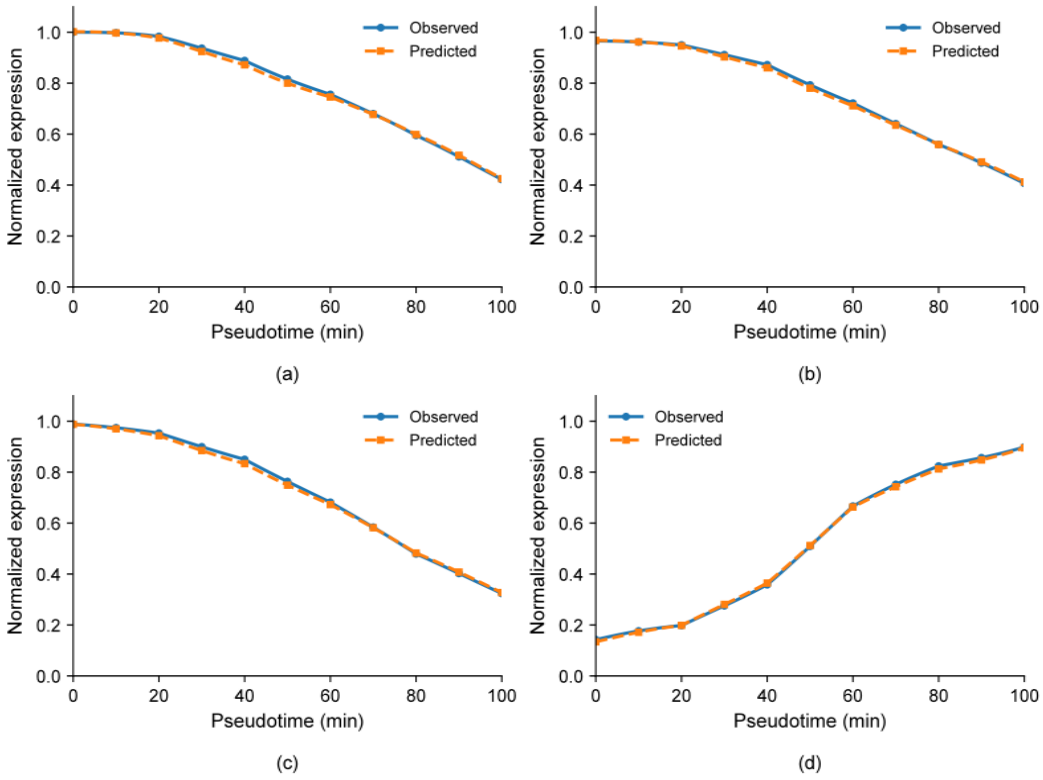

Figure 3: Comparison of actual and predicted expression trajectories.

In Fig. 3, in the latter segment of the pseudo-time series, POU5F1, SOX2 and NANOG show continuous decline, whereas GATA6 has a quick rise in its middle and early stages. The inflection points and change rates of the predicted and observed curves present a very high consistency; therefore, the Pearson correlation coefficient for GATA6 arrives at 0.944, and the RMSE for NANOG is kept at 0.073, hence indicating that the model has a strong capability to reconstruct both up-regulating and down-regulating dynamic processes. To further check whether the model can catch continuous state transitions during branch differentiation, Fig. 4 provides the latent space phase diagram.

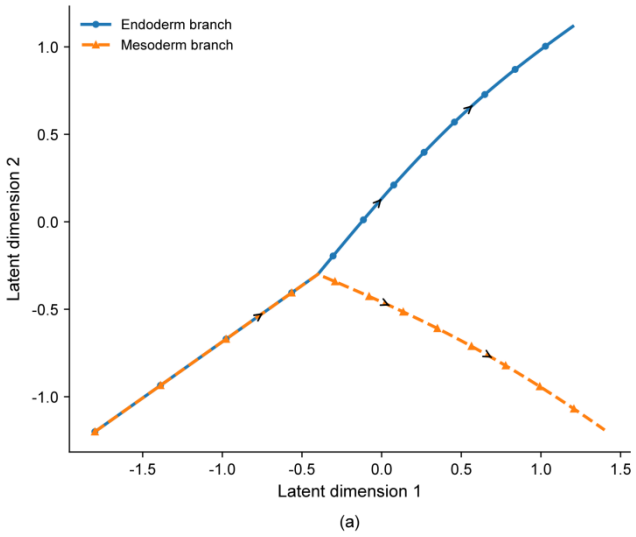

Figure 4: Phase diagram of latent space differentiation trajectories

In Fig. 4, the two main branches hold a same starting state in the early area, after that step by step separating and showing a clear split in the pseudo-time period from 55 to 70 minutes. The average track difference for the endoderm branch is 0.118, and 0.126 for the mesoderm branch; hence both numbers are lower than the control model's 0.150 level. The track shapes in Fig. 4 show that the change of hidden states did not have local backward moves or sudden gathering; therefore the space between branches became bigger little by little as time passed, thus suggesting that the ODE flow field has a relatively steady and non-broken ability to describe differentiation tracks. GRLGRN lays strong stress on using pre-existing graphs and graph representation study to get back hidden regulatory relationships, which is good for structural recovery [24]; in the current task, continuous-time integration further raises the expression strength about the order of branch development, so it presents a more smooth state spread at the phase space level. Fig. 5 gives the heatmap results of the guessed adjacency matrix on typical regulatory modules.

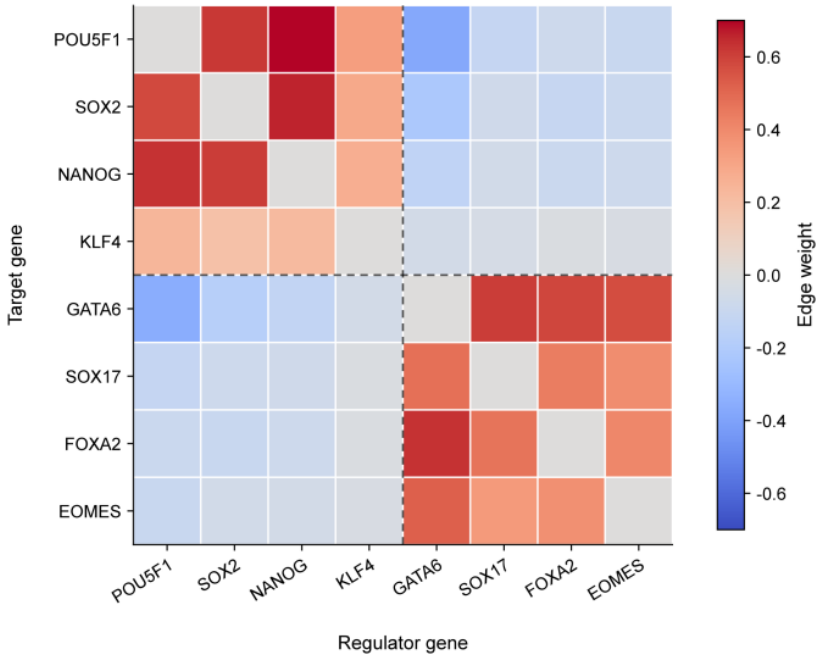

Figure 5: Heatmap of the inferred adjacency matrix for representative regulatory modules.

In Fig. 5, strong positive regulatory weights are observed between POU5F1, SOX2 and NANOG, ranging from 0.54 to 0.72; the activation weights of GATA6 for SOX17, FOXA2 and EOMES range from 0.57 to 0.63; simultaneously, the inhibitory weights of GATA6 on POU5F1 and POU5F1 on GATA6 are -0.36 and -0.38, respectively. These results are consistent with the overall pattern observed during germ layer differentiation, wherein pluripotency modules gradually weaken whilst endoderm-related modules continue to strengthen. When analysed based on the top 20 high-confidence edges, the proportion of activation/inhibition directions consistent with known developmental relationships was 85.0%, indicating that the model's output not only provides connection strengths but also offers a relatively clear explanatory basis at the directional level. To observe differences in the temporal responses of key regulatory factors, Fig. 6 plots the temporal response curves of the top 5 regulators.

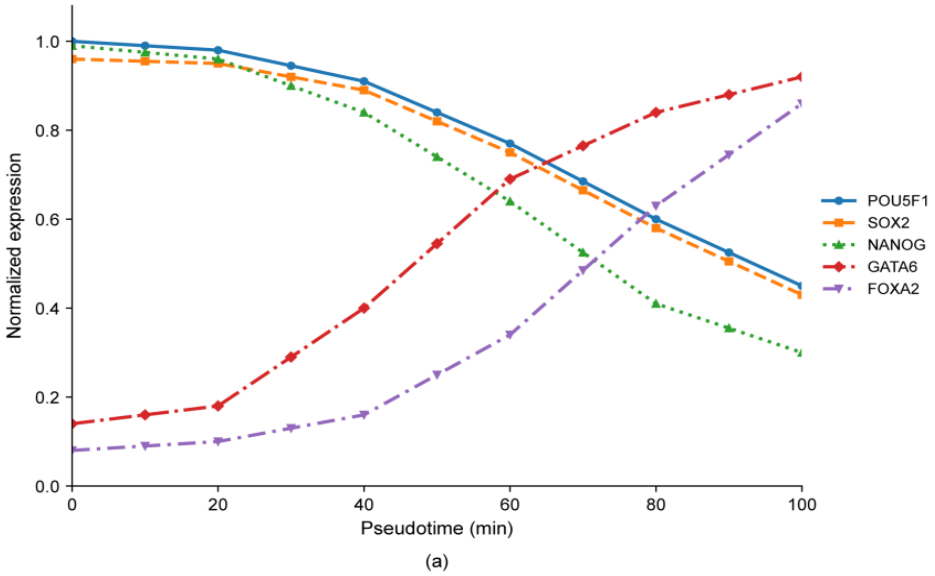

Figure 6: Time-response curves of key regulatory factors.

In Fig. 6, POU5F1, SOX2 and NANOG decrease at the same time after 40 minutes, with NANOG having the largest reduction, dropping to 0.46 times from the initial level to the final point; GATA6 ascends most quickly between 35 and 60 minutes, and its peak slope appears at about 47 minutes; FOXA2 starts a little later, entering a fast-increase stage around 60 minutes. Therefore, when we combine it with Fig. 5, we can find that the POU5F1/SOX2/NANOG module, the GATA6/FOXA2/SOX17 module, and the KLF4/DPPA4 module show a time-sequential relationship, thus existing a clear window phase between the weakening of the former module and the strengthening of the latter one. Hence, this shows that the regulatory network rebuilt by the model provides a firm foundation for the continuous reconstruction of real expression paths, meanwhile it also keeps biological interpretability that is suitable for follow-up mechanism analysis.

3.3 Ablation analysis and limitations of the proposed framework

The ablation results are shown in Fig. 7. The full model maintains the best performance across all three network scales. Taking the 100-gene network as an example, the full model achieves AUROC and AUPR values of 0.872 and 0.631, respectively; after removing the structural sparsity constraint, these figures drop to 0.846 and 0.592, further dropping to 0.833 and 0.574 after removing the temporal consistency loss, and to 0.838 and 0.581 after removing the biological prior; when the Neural ODE was replaced with a discrete RNN/MLP update, these figures fell to just 0.821 and 0.547. For the 200-gene network, this difference is even more pronounced: the AUPR of the full model is 0.556, whilst the four ablation settings yield 0.518, 0.503, 0.509 and 0.472 respectively. As can be seen from Figures 7(a) and 7(b), the temporal consistency loss and continuous-time updates have the greatest impact on performance, indicating a strong coupling between structural identification quality and state-advancement constraints; the contributions of sparse constraints and biological priors are primarily reflected in suppressing redundant edges and stabilising the high-dimensional candidate space.

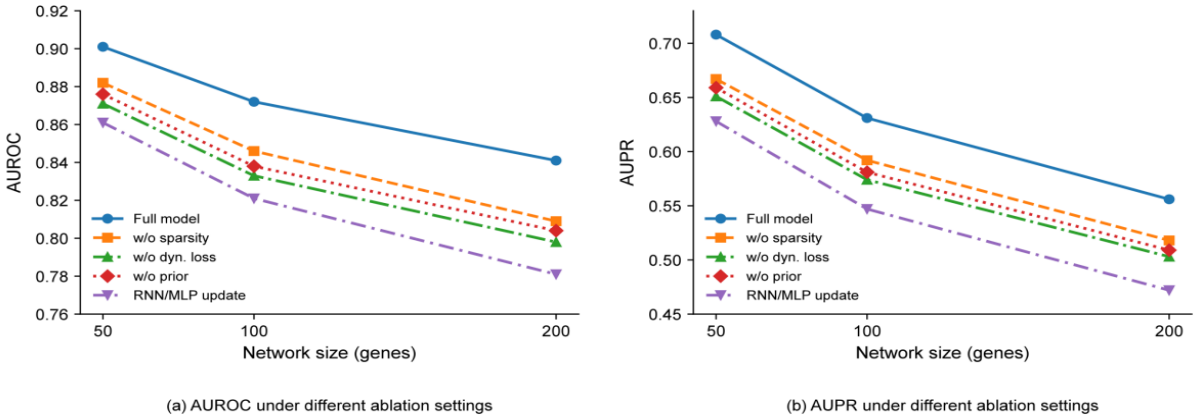

Figure 7: Variation in structural recognition performance under different ablation settings.

The trends shown in Fig. 7 are consistent with those observed on real expression data. In the hESC dataset, the full model achieved an RMSE of 0.084 and a trajectory correlation of 0.928; after removing the temporal consistency loss, the RMSE rose to 0.113 and the correlation coefficient fell to 0.891; after removing the biological prior, these values became 0.101 and 0.907 respectively; after removing the sparsity constraint, they were 0.097 and 0.911; when using discrete RNN/MLP updates, the RMSE reached 0.121 and the correlation coefficient dropped to 0.883. The trends in the mESC dataset are similar: the correlation coefficient for the full model is 0.914, whilst the corresponding values for the four ablation groups are 0.901, 0.879, 0.896 and 0.871. Combining these results, it can be seen that the temporal consistency loss plays a key role in linking structural learning with trajectory reconstruction; without this term, although the model can still produce usable edge ordering, the accuracy of dynamic fitting declines more rapidly. Fig. 8 further illustrates the changes in computational cost and high-dimensional scaling.

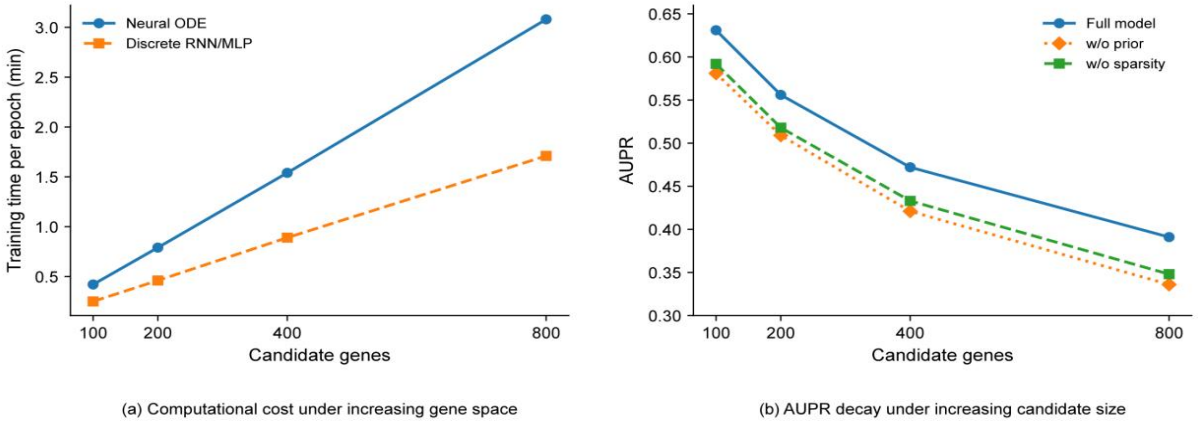

Figure 8: Analysis of model efficiency and high-dimensional scalability

In Fig. 8, when candidate gene number goes up from 100 to 800, per-epoch training time for Neural ODE version climbs from 0.42 min to 3.08 min, whereas that for discrete RNN/MLP version increases from 0.25 min to 1.71 min; the former is about 1.7 to 1.8 times of the latter. The growth of training cost comes from extra overhead of continuous integration and related differentiation, but corresponding structural recovery quality is more stable. In the same scaling process, AUPR of full model dropped from 0.631 to 0.391; when biological prior was removed, it dropped from 0.581 to 0.336; when sparsity constraint was removed, it

dropped from 0.592 to 0.348. Therefore, this shows that as candidate gene space keeps expanding, adjacency matrix learning faces notable search pressure; hence, prior constraints and sparsity constraints can reduce performance decline to some degree, but they cannot fully counteract uncertainty brought by high-dimensional search.

About limitations, current framework still needs further polish in three aspects. Firstly, continuous-time modelling strengthens dynamic expression ability, but training efficiency is still lower than discrete-update models, thus leading to higher computation and memory burdens on larger-scale networks. Secondly, candidate edge number increases too fast in ultra-high-dimensional gene spaces; only depending on expression data and sparsity regularisation for filtering still causes false positive accumulation and local optimum trapping problems. Thirdly, when only relying on transcriptomic information, edge direction and interaction sign determination is still restricted by covariance patterns; especially when feedback loops and parallel regulation exist together, some high-confidence edges still need additional evidence for support. Existing research has started to integrate heterogeneous graphs with Transformers into single-cell GRN inference to enhance cross-modal relationship representation and long-range dependency modelling capabilities [25]. Continuing along this direction by adding ATAC, ChIP or multi-omics prior information is expected to further raise reliability of edge direction identification and module interpretation.

4 Conclusion

(1) We have developed a combined Neural ODE-GRN framework which works for continuous-time expression data, and we integrate adjacency matrix studying, hidden state development and observation rebuilding into one single optimization procedure; therefore, this makes the rebuilding of regulatory edges and the characterization of expression dynamics be carried out in a cooperative way inside a unified model.

(2) This framework demonstrates robust advantages across simulated networks of 50, 100 and 200 genes, as well as on real-world hESC and mESC data: under 100-gene conditions, the AUPR is improved to 0.631, AUROC remains above 0.80 under 20% noise, and trajectory correlation on real data reaches 0.914–0.928, whilst maintaining good topological consistency for hub nodes and key regulatory modules.

(3) The results indicate that this method provides a continuous, interpretable analytical tool for elucidating regulatory mechanisms in developmental differentiation and disease progression; further refinements are required regarding search efficiency in high-dimensional sparse scenarios, methods for incorporating cross-modal prior knowledge, and the computational cost of training large-scale networks.

Author's Profile

Song Kangsheng was born in Yantai City, Shandong Province, China in 2000. I graduated from Shandong University of Technology and obtained a bachelor's degree. Currently, I am pursuing a master's degree at the School of Computer Science and Technology of Shandong University of Technology. My main research field is bioinformatics. 19862520287@163.com

Liu Qiankun was born in Zhoukou City, Henan Province, China in 1988. I graduated from Linyi University and obtained a bachelor's degree. Currently, I am pursuing a master's degree at the School of Computer Science and Technology of Shandong University of Technology. My main research fields are big data analysis and bioinformatics. dabaiyanglqk@163.com

Gao Zhihong was born in Luliang City, Shanxi Province, China in 2000. I graduated from

Shanxi Institute of Engineering Technology and obtained a bachelor's degree. Currently, I am pursuing a master's degree at the College of Computer Science and Technology of Shandong University of Technology. My main research fields are big data analysis and bioinformatics. gaozhihong73@163.com

References

[1] Badia-I-Mompel P, Wessels L, Müller-Dott S, et al. Gene regulatory network inference in the era of single-cell multi-omics. Nature Reviews Genetics, 2023, 24(11): 739-754. DOI: 10.1038/s41576-023-00618-5.

[2] Kim D, Tran A, Kim H J, et al. Gene regulatory network reconstruction: harnessing the power of single-cell multi-omic data. npj Systems Biology and Applications, 2023, 9(1): 51. DOI: 10.1038/s41540-023-00312-6.

[3] Ali S I M, Alrashid S, et al. A review of methods for gene regulatory networks reconstruction and analysis. Artificial Intelligence Review, 2025, 58: 256. DOI: 10.1007/s10462-025-11257-z.

[4] Keyl P, Bischoff P, Dernbach G, et al. Single-cell gene regulatory network prediction by explainable AI. Nucleic Acids Research, 2023, 51(4): e20. DOI: 10.1093/nar/gkac1212.

[5] Li L, Sun L, Chen G, et al. LogBTF: gene regulatory network inference using Boolean threshold network model from single-cell gene expression data. Bioinformatics, 2023, 39(5): btad256. DOI: 10.1093/bioinformatics/btad256.

[6] Zeng Y, He Y, Zheng R, Li M. Inferring single-cell gene regulatory network by non-redundant mutual information. Briefings in Bioinformatics, 2023, 24(5): bbad326. DOI: 10.1093/bib/bbad326.

[7] Xu J, et al. STGRNS: an interpretable transformer-based method for inferring gene regulatory networks from single-cell transcriptomic data. Bioinformatics, 2023, 39(4): btad165. DOI: 10.1093/bioinformatics/btad165.

[8] Mao G, Pang Z, Zuo K, et al. Predicting gene regulatory links from single-cell RNA-seq data using graph neural networks. Briefings in Bioinformatics, 2023, 24(6): bbad414. DOI: 10.1093/bib/bbad414.

[9] Wang J, Chen Y, Zou Q. Inferring gene regulatory network from single-cell transcriptomes with graph autoencoder model. PLoS Genetics, 2023, 19(9): e1010942. DOI: 10.1371/journal.pgen.1010942.

[10] Wang Y, Chen X, Zheng Z, et al. scGREAT: Transformer-based deep-language model for gene regulatory network inference from single-cell transcriptomics. iScience, 2024, 27(4): 109352. DOI: 10.1016/j.isci.2024.109352.

[11] Sha Y, Qiu Y, Nie Q. NeuralGene: Inferring gene regulation and cell-fate dynamics from Neural ODEs. Journal of Machine Learning for Modeling and Computing, 2023, 4(3): 1-15. DOI: 10.1615/JMachLearnModelComput.2023047369.

[12] Farrell S, Mani M, Goyal S. Inferring single-cell transcriptomic dynamics with structured latent gene expression dynamics. Cell Reports Methods, 2023, 3(9): 100581. DOI: 10.1016/j.crmeth.2023.100581.

[13] Li Q. scTour: a deep learning architecture for robust inference and accurate prediction of cellular dynamics. Genome Biology, 2023, 24(1): 149. DOI: 10.1186/s13059-023-02988-9.

[14] Yuan Q, Duren Z, et al. Inferring gene regulatory networks from single-cell multiome data using atlas-scale external data. Nature Biotechnology, 2025, 43(2): 247-257. DOI: 10.1038/s41587-024-02182-7.

[15] Li S, Liu Y, Shen L C, et al. GMFGRN: a matrix factorization and graph neural network approach for gene regulatory network inference. Briefings in Bioinformatics, 2024, 25(2): bbad529. DOI: 10.1093/bib/bbad529.

[16] Yuan L, Zhao L, Jiang Y, et al. scMGATGRN: a multiview graph attention network-based method for inferring gene regulatory networks from single-cell transcriptomic data. Briefings in Bioinformatics, 2024, 25(6): bbae526. DOI: 10.1093/bib/bbae526.

[17] Zhang Y, Wang M, Wang Z, et al. MetaSEM: Gene Regulatory Network Inference from Single-Cell RNA Data by Meta-Learning. International Journal of Molecular Sciences, 2023, 24(3): 2595. DOI: 10.3390/ijms24032595.

[18] Wang J-C, Chen Y-J, Zou Q. GRACE: Unveiling Gene Regulatory Networks With Causal Mechanistic Graph Neural Networks in Single-Cell RNA-Sequencing Data. IEEE Transactions on Neural Networks and Learning Systems, 2025, 36(5): 9005-9017. DOI: 10.1109/TNNLS.2024.3412753.

[19] Lamoline F, et al. Dynamic gene regulatory network inference from single-cell data with GRIT. Bioinformatics, 2025, 41(8): btaf394. DOI: 10.1093/bioinformatics/btaf394.

[20] Jiang H, Wang Y, Yin C, et al. SLIVER: Unveiling large scale gene regulatory networks of single-cell transcriptomic data through causal structure learning and modules aggregation. Computers in Biology and Medicine, 2024, 178: 108690. DOI: 10.1016/j.compbiomed.2024.108690.

[21] Gan Y, Yu J, Xu G, Yan C, Zou G. Inferring gene regulatory networks from single-cell transcriptomics based on graph embedding. Bioinformatics, 2024, 40(5): btae291. DOI: 10.1093/bioinformatics/btae291.

[22] Stock M, Popp N, Fiorentino J, et al. Topological benchmarking of algorithms to infer gene regulatory networks from single-cell transcriptomic data. Bioinformatics, 2024, 40(5): btae267. DOI: 10.1093/bioinformatics/btae267.

[23] Zhu H, Slonim D K. Improved gene regulatory network inference from single cell data with dropout augmentation. PLoS Computational Biology, 2025, 21(10): e1013603. DOI: 10.1371/journal.pcbi.1013603.

[24] Wang K, Li Y, Liu F, et al. GRLGRN: graph representation-based learning to infer gene

regulatory networks from single-cell RNA-seq data. BMC Bioinformatics, 2025, 26(1): 108. DOI: 10.1186/s12859-025-06116-1.

[25] Sun Y, Gao J. HGATLink: single-cell gene regulatory network inference via the fusion of heterogeneous graph attention networks and transformer. BMC Bioinformatics, 2025, 26(1): 49. DOI: 10.1186/s12859-025-06071-x.