Reviewed Preprint
v1 • November 25, 2025
Not revised

Reviewed Preprint
v2 • June 22, 2026
Revised by authors

For correspondence:
yyuan@ipe.ac.cn

Competing interests: No competing interests declared

Funding: See page 19

Reviewing editor: Yongliang Yang, Shanghai University of Medicine and Health Sciences, China

TSvelo: Comprehensive RNA velocity by modeling the cascade of gene regulation, transcription and splicing

Jiachen Li[1], Zhe Wang[1], Hong-Bin Shen[2], Ye Yuan[1,2]

[1]State Key Laboratory of Biopharmaceutical Preparation and Delivery, Institute of Process Engineering, Chinese Academy of Sciences, Beijing, China • [2]Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, China

eLife Assessment

This study presents a **valuable** RNA velocity method which predicts the transcription rate linearly based on the expression of RNA levels of transcription factors with addition of comprehensive analyses. The evidence supporting the claims of the authors is **solid**, although inclusion of a full simulation would have strengthened the study. The work will be of interest to scientists working in the field of RNA biology and precision medicine.

https://doi.org/10.7554/eLife.108950.2.sa3

Abstract

RNA velocity approaches fit gene dynamics and infer cell fate by modeling the splicing process using single-cell RNA sequencing (scRNA-seq) data. However, due to short time scale of splicing, high noise and large complexity of data, existing RNA velocity methods often fail to precisely capture the complex velocity dynamics for individual gene and single cell, which makes its downstream analysis less reliable and less robust. We propose **TSvelo**, a comprehensive RNA **velo**city mathematics framework that can model the cascade of gene regulation, **T**ranscription and **S**plicing using highly interpretable neural Ordinary Differential Equations (ODEs). TSvelo can precisely capture the transcription-unspliced-spliced 3D dynamics of all genes simultaneously, infer unified latent time shared by genes within single cell, and be applied to multi-lineage datasets. Experiments on six scRNA-seq datasets, including two multi-lineage datasets, demonstrate TSvelo's superiority.

Introduction

Single-cell RNA sequencing (scRNA-seq) enables the detailed exploration of gene expression at the individual cell level. To move beyond static snapshots and capture the dynamic behavior of cells over time, several trajectory inference (TI) methods have been developed, such as PAGA[1], Monocle[2], Slingshot[3] and Palantir[4]. These methods typically estimate pseudotime using diffusion processes and require prior annotation of initial cells. In contrast, RNA velocity[5] offers a more interpretable approach by modeling the time derivative of gene expression, linking unspliced (immature) and spliced (mature) mRNA levels through Ordinary Differential Equations (ODEs).

Several RNA velocity methods have been proposed to capture splicing dynamics. The first approach, Velocyto[5], uses least squares solutions to estimate parameters under the assumption of steady-state kinetics. ScVelo[6] improves upon this by employing an Expectation-Maximization (EM) approach for better fitting splicing kinetics.

UniTVelo[7] propose a top-down idea that directly models the spliced RNA levels with a time-dependent function. More recently, generative models such as VeloVI[8], veloVAE[9], Pyrovelocity[10] and BayVel[11] have been introduced, utilizing Bayesian frameworks to estimate RNA velocity. To handle multi-lineage datasets, methods like CellDancer[12], DeepVelo[13] and LatentVelo[14] extend RNA velocity by modeling local dynamics with neural networks, rather than assuming a globally constant transcriptional rate. Apart from using unspliced/spliced data, Dynamo[15] enhances RNA velocity further by incorporating labeled RNA-seq data. The advent of single-cell multi-omics[16,17] technologies has allowed RNA velocity analysis to extend to protein abundance (e.g., protaccel[18]) and single-cell ATAC-seq datasets (e.g., MultiVelo[19]). Additionally, methods like DeepCycle[20] and VeloCycle[21] have been developed to focus on cell cycle processes, with specialized modules for capturing periodic signals. STT[22] and SIRV[23] proposed the idea on extending RNA velocity analysis to spatial transcriptomics.

Although RNA velocity theory has significantly advanced the inference of single-cell trajectories, pseudotime, and gene regulation[24], several challenges persist for current RNA velocity models. Firstly, RNA velocity models primarily infer cell fate based on phase portrait fitting of unspliced and spliced dynamics for each gene. However, they often fail to capture the correct phase portrait for most genes, due to the sparsity and noise in unspliced and spliced mRNA abundance for individual genes, the short time scale of the splicing process, and the mixing of cells from different types on the phase portrait[25-27]. Secondly, The majority of existing RNA velocity models treat each gene independently and fail to incorporate the underlying regulatory interactions[28]. Although some approaches (e.g. TFvelo[27], PHOENIX[29], scKINETICS[30] and scPN[31]) have constructed ODE models to capture gene dynamics by integrating regulation information, these methods overlook the splicing signal so that they cannot jointly model the transcription and splicing dynamics into one unified form. Thirdly, classical RNA velocity approaches, such as scVelo, use interpretable parameters in constructing dynamic model for single gene. By contrast, to model flexible transcriptional rates or integrate multiple genes, several recent methods employ latent space embeddings or neural network-based encoders[12]. It makes the model parameters less interpretable in detailed gene level, which is crucial for understanding the underlying biological mechanisms. Fourthly, multi-lineage tasks remain a significant challenge for current RNA velocity models due to the complexity in large scale scRNA-Seq datasets.

To address the challenges outlined above, we propose TSvelo, a method that intergrates gene regulation, transcription and splicing of all genes into a single ODE model, whose parameters are highly interpretable. Using a high-dimentional Neural ODE solver[32,33], TSvelo directly learns the global latent time without the need to separately learn gene-specific latent times. By leveraging both unspliced and spliced scRNA-seq data, along with gene regulatory knowledge from TF-target databases, TSvelo iteratively optimizes the parameters in the ODE model and the unified latent time using Expectation-Maximization (EM) algorithm. Experiments on six scRNA-seq datasets, including two multi-lineage datasets, demonstrate that TSvelo outperforms existing methods in modeling gene dynamics, inferring cell fates, and is also effective in analyzing multi-lineage datasets.

Results

Estimate RNA velocity with TSVelo

We present TSvelo, a method for jointly estimating RNA velocity across high-dimentional genes by integrating both transcriptional regulation and splicing information. Existing RNA velocity models predominantly focus on modeling the splicing process and typically operate in a gene-wise manner. In contrast, TSvelo explicitly models the full cascade of regulation, transcription, and splicing, while capturing their coordinated dynamics across all genes simultaneously. A systematic comparison between TSvelo and representative RNA velocity methods is provided in Table S1 at Supplementary Information. In TSvelo, the unspliced and spliced RNA abundances are initially preprocessed for velocity gene selection and pseudotime initialization (Fig. 1a). See Methods for

details about preprocessing. Next to model the velocity gene g's expression dynamics, we suppose u_g and s_g are the abundance of unspliced and spliced RNA, and $\alpha_g(t)$, β_g,γ_g are the transcription rate, splicing rate and degradation rate, respectively. The dynamics is modeled as:

$$\frac{du_g(t)}{dt} = \alpha_g(t) - \beta_g u_g(t) \tag{1}$$

$$\frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t) \tag{2}$$

Furthermore, we assume that the gene and cell-specific transcriptional rate $\alpha_g(t)$ is influence by the expression of Transcriptional Factors (TFs).

Considering the wide usage and interpretability of linear models for gene-relations in previous studies[27,34-37], we model $\alpha_g(t)$ as

$$\alpha_g(t) = \mathrm{ReLU}\left(\sum_{i \in TFS(g)} w_{gi} * s_i(t)\right) \tag{3}$$

ReLU denotes the Rectified Linear Unit activation function, defined as $ReLU(x) = \max(0, x)$. The term w_{gi} represents the regulatory weight of TF i on the target gene g. $TFs(g)$ refers to the TFs that could regulated the target gene g, which are selected according to the ChEA and ENCODE TF-target database. In order to include more TFs in the model, we also reserve those TFs that are not selected as velocity genes, and directly model the dynamics between transcription and spliced RNA without using the unspliced abundance. Finally, we combine the dynamic models on all selected genes into the Ordinary Differential Equation (ODE) matrix form (Fig. 1b), which enables directly inferring a unified latent time for each cell using its all genes. See Methods for details.

TSvelo employs an EM framework to iteratively optimize both latent time t and the parameters in ODE. The global pseudotime t is assigned to each cell through grid search. Due to the difficulty in calculating analytical solutions of $u_g(t)$ or $s_g(t)$, TSvelo adopted a Neural ODE for estimating those parameters of transcription rate, splicing rate and degradation rate. Next, TSvelo can model the gene dynamics of both transcription, unsplicing and splicing processes (Fig. 1c), predict cell states using both pseudotime and velocity stream analysis (Fig. 1d), and is also applicable to multi-lineage tasks for analyzing expression patterns across different lineages in large complex scRNA-Seq datasets (Fig. 1e). A validation of TSvelo on simulated data is provided in Fig. S1 and Fig. S2 in the Supplementary Information.

TSvelo can model 3D gene dynamics and predict cell fate on pancreas dataset

To validate the TSvelo model, we applied it to the pancreas scRNA-seq dataset[5], which is widely used in RNA velocity studies to model cell differentiation from ductal to endocrine cells. Using the predicted RNA velocity, TSvelo can generate velocity stream plot using the functions provided in scVelo as well. Both the pseudotime t (Fig. 2a) and the velocity stream plot (Fig. 2b) effectively capture the cell differentiation process. We quantitively compare TSvelo and baseline approaches, including scVelo[6], dynamo[15], UniTVelo[7], cellDancer[12] and TFvelo[27].

In the conventional 2D unspliced–spliced phase portrait, cells from different clusters often overlap, limiting separability. By introducing the latent variable α, the representation is extended to a 3D space, which helps disentangle these mixed states and reveals a clearer phase structure. As shown in Fig. 2c, the 3D representation achieves consistently higher kNN classification accuracy for cell-state separation than the 2D u–s embedding (one-sided Mann–Whitney U test, p = 4.37 ×

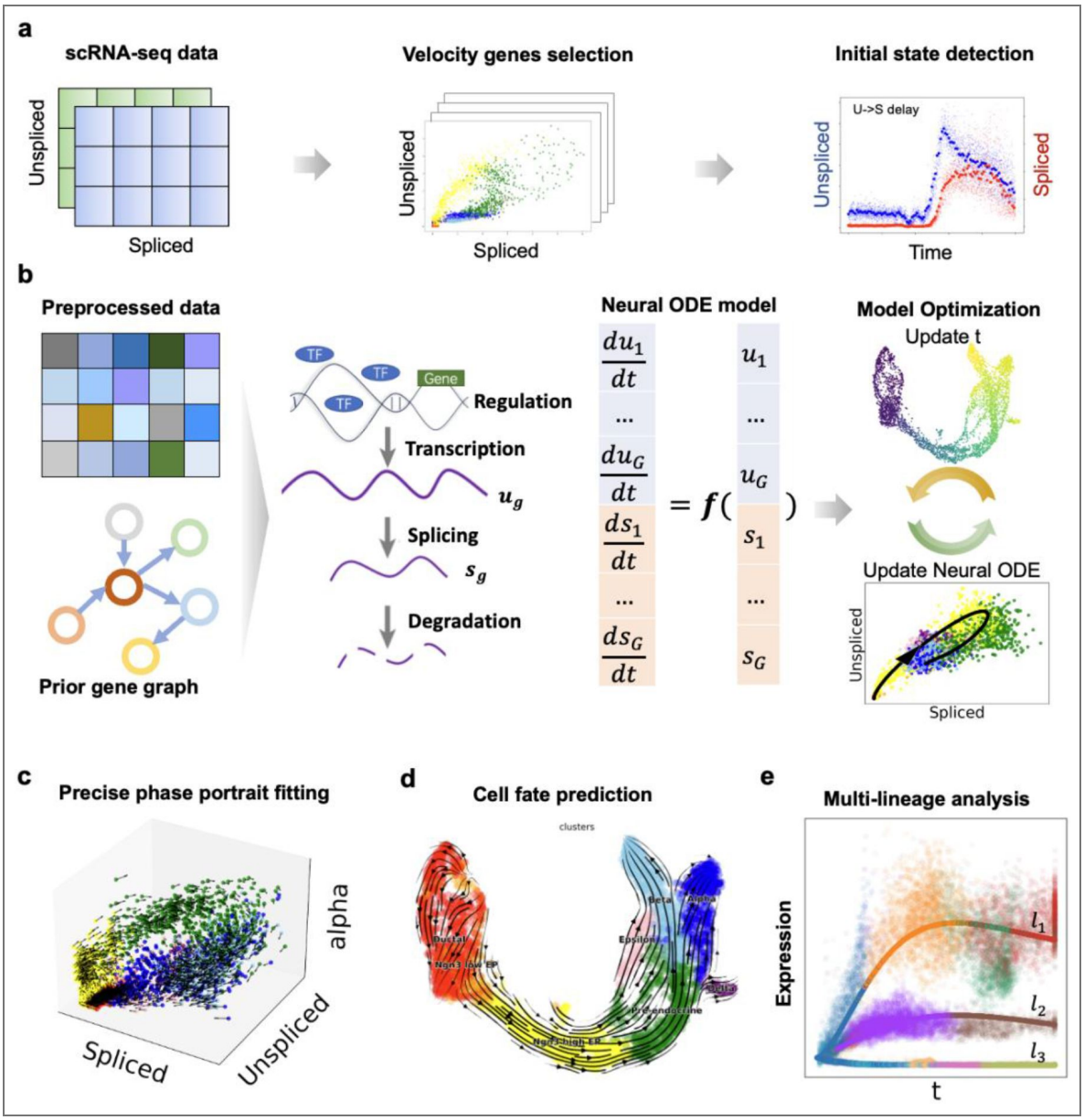


Figure 1. The framework of TSvelo.

(**a**) The preprocessing strategy in TSvelo, including velocity genes selection and initial state detection. (**b**) The Neural ODE model and its optimization in TSvelo, where the parameters and latent time is optimized iteratively. (**c-e**) The downstream application tasks of TSvelo, including precise transcription-unspliced-spliced 3D phase portrait fitting (c), cell fate prediction using predicted RNA velocity (d) and gene expression pattern analysis for multi-lineage dataset (e).

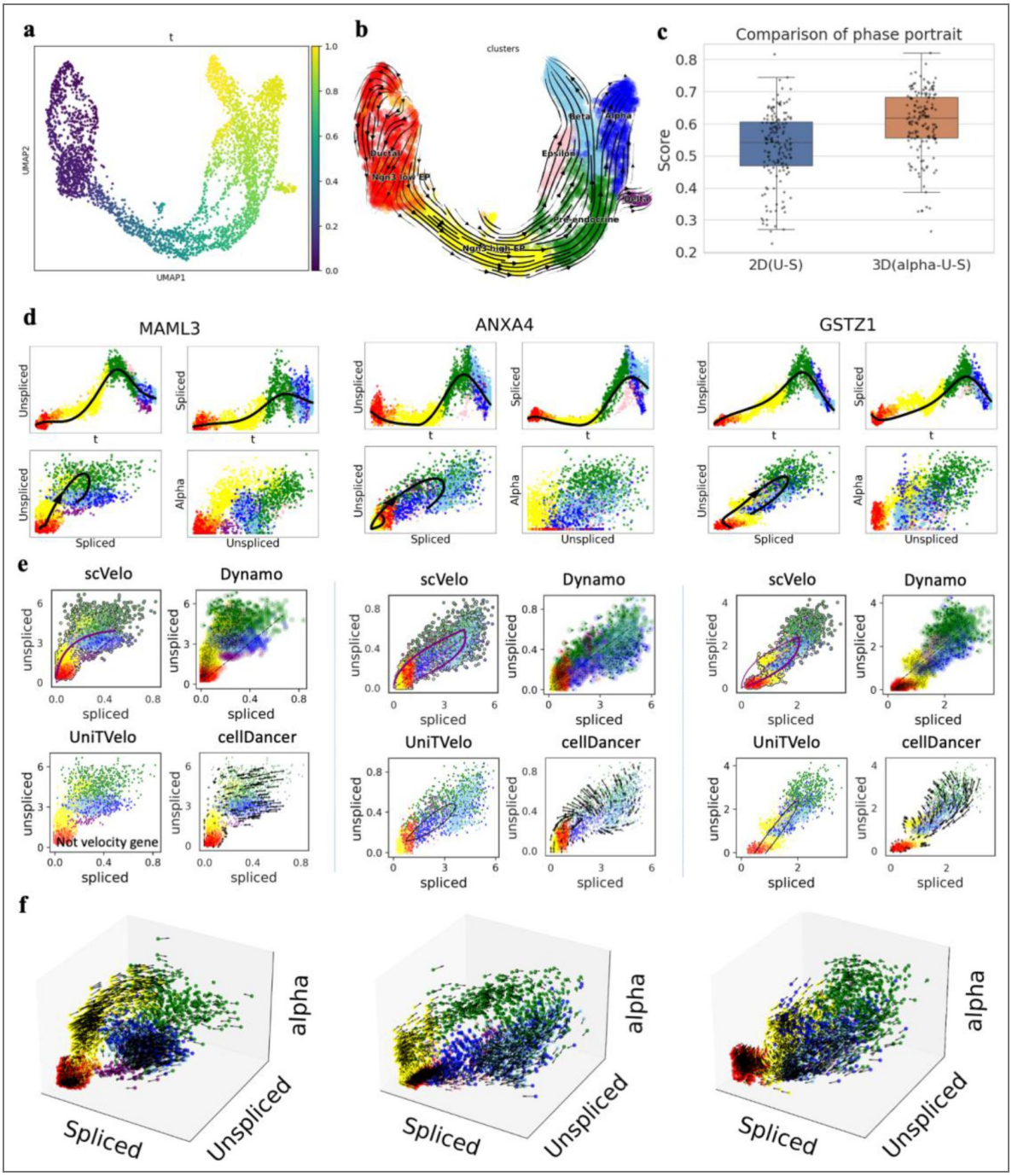


Figure 2. **Results on pancreas dataset.**

(**a**) The pseudotime learned with TSvelo. (**b**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. (**c**) A quantitative comparison of cell-state separability between the 3D phase portrait used in TSvelo and the traditional 2D phase portrait. The down, central and up hinges correspond to the first quartile, median value and third quartile, respectively. The whiskers extend to 1.5× the interquartile range of the distribution from the hinge. 3,696 samples are included in the boxplot for each method. (**d**) The dynamics fitting on MAML3, ANXA4 and GSTZ1. For each gene, four plots are displayed in a 2×2 layout: the u-t, s-t, u-s, and alpha-u plots. (**e**) The unspliced-spliced phase portrait fitting on MAML3, ANXA4 and GSTZ1 obtained by four baseline RNA velocity approaches, including scVelo, Dynamo, UniTVelo and cellDancer. (**f**) The dynamics fitting in transcription-unspliced-spliced 3D phase portrait for MAML3, ANXA4 and GSTZ1.

10^{-10}. Detials are provided in Methods). These results indicate that the 3D phase portrait provides improved separation of cell states and a stronger foundation for modeling the underlying dynamics along cell trajectories.

Additional quantitative evaluation is provided in Fig. S4 at Supplementary Information). TSvelo achieves the highest median velocity consistency, which demonstrates that the high-dimensional velocity vectors learned by TSvelo are mostly coherent within neighbor cells. TSvelo also achieves the highest median in-cluster coherence and cross-boundary correctness, which validates that TSvelo best fit the differentiation process within these cell types according to the ground truth annotation.

Next, we show the dynamics for individual genes and demonstrate how the TSvelo model aids gene dynamics analysis by incorporating both transcriptional and splicing information. MAML3, ANXA4 and GSTZ1 are selected as examples because their unspliced–spliced 2D phase portrait exhibits mixed or overlapping patterns that are difficult to model using conventional RNA velocity approaches. Fig. 2d present the results of the TSvelo model on these genes, and Fig. 2e present the results of baseline approaches. Many previous approaches assume that the unspliced-spliced phase portrait exhibits an almond-shaped distribution[38]which may not actually hold in the data. For MAML3, while the unspliced-spliced distribution predominantly follows the almond-shaped pattern, some cells, specifically Alpha cells (blue) and Beta cells (light blue), overlap with Ngn3 high EP cells (yellow), indicating that these cell types cannot be distinctly separated using only the unspliced-spliced 2D phase portrait. Nevertheless, TSvelo can infer transcription representation from the expression of multiple transcription factors, which enables the model to distinguish these cell types (Fig. 2d and Fig. 2f). ANXA4 shows higher expression in Ductal cells (in red) compared to Ngn3 low EP cells (in orange), which mean its expression pattern exhibits an initial decrease followed by an increase. Such dynamics are not easily captured in the conventional unspliced–spliced phase portrait used by previous approaches, as many baseline methods implicitly assume a decreasing–then–increasing expression pattern. By comparison, TSvelo can still fit such expression pattern by using additional information from the 3D phase portrait. The visualization of dynamics fitting on more genes are provided at Fig. S5 and Fig. S6 in Supplementary Information.

TSvelo can better predict RNA velocity on gastrulation erythroid dataset

We applied TSvelo to the gastrulation erythroid dataset, which is derived from the transcriptional profile of mouse embryos [39] and has been used in previous RNA velocity studies. This dataset primarily describes the differentiation process from blood progenitors to erythroid cells. First, using Gene Ontology (GO) term enrichment analysis (Fig. 3a), we find that the velocity genes selected through TSvelo's preprocessing strategies are mostly enriched in the to the erythropoiesis-related process, providing a preliminary validation of the biological plausibility of the inferred velocities.

After applying TSvelo, both the inferred pseudotime (Fig. 3b) and the velocity stream plot (Fig. 3c) are compatible with current biological knowledge about the cell differentiation process. We also compare TSvelo with previous approaches. The results, shown in Fig. 3d-f, demonstrate that TSvelo achieves the highest velocity consistency, the highest in-cluster coherence, and the highest cross-boundary correctness, showing that the high-dimensional dynamics predicted by TSvelo best capture the underlying biological processes.

We next analyze TSvelo modeling at the gene level in Fig. 3g-i. The 3D phase portrait again provides a better basis for dynamic fitting, as it more effectively separates cells from different clusters than the traditional 2D phase portrait (Fig. S7). For genes exhibiting higher noise or more complex patterns, TSvelo demonstrates its robustness in dynamic fitting by integrating both transcriptional and splicing signals. For HSP90AB1, which exhibits a counter-clockwise pattern in the unspliced–spliced phase portrait (Fig. 3g), in contrast to the clockwise dynamics typically assumed by most baseline approaches, it is difficult for previous methods to capture this behavior

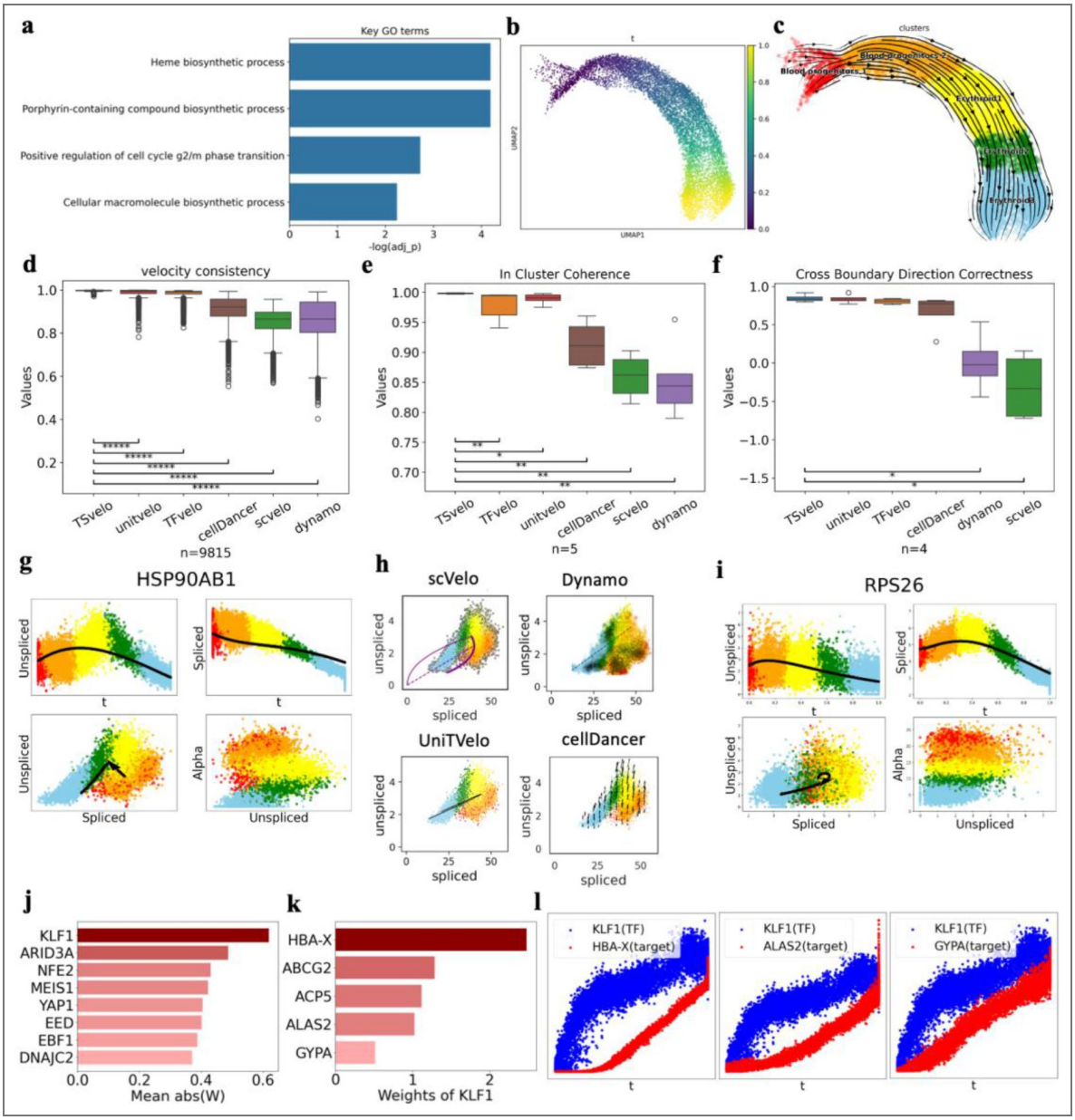


Figure 3. Results on gastrulation erythroid dataset.

(**a**) The GO terms which are mostly enriched in the selected velocity genes of TSvelo. (**b**) The pseudotime learned with TSvelo. (**c**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. (**d-f**) The quantitative comparison between TSvelo and multiple baseline approaches in terms of velocity consistency (d), in-cluster coherence (e), and cross boundary direction correctness (f). The down, central and up hinges correspond to the first quartile, median value and third quartile, respectively. The whiskers extend to 1.5× the interquartile range of the distribution from the hinge. 9,815 samples are included in the boxplot for each method. (**g**) The dynamics fitting on HSP90AB1 obtained by TSvelo. Four plots are displayed in a 2×2 layout: the u-t, s-t, u-s, and alpha-u plots, where u, a and alpha mean the abundance of unspliced mRNA, the abundance of spliced mRNA, and the learned transcriptional representation, respectively. (**i**) The phase portrait fitting on HSP90AB1 obtained by baseline approaches. (**i**) The dynamics fitting on RPS26 obtained by TSvelo. (**j**) The TFs with the highest ranked weights as identified by TSvelo. (**k**) Weights for KLF1's targets, with the highest absolute weight. (**l**) Temporal dynamics of KLF1 and its target genes with the highest weights along pseudotime, which includes HBA-X, ALAS2 and GYPA.

(Fig. 3h), whereas TSvelo can still faithfully model such patterns. For genes such as RPS26, which have critical roles in the development in blood progenitors to erythroid[40], the unspliced-spliced data is so noisy that cells of different types overlap in phase portrait. TSvelo can still captures the gene dynamics and reveals differences in transcription rates (Alpha) across cell types (Fig. 3i). In contrast, methods that rely solely on unspliced-spliced data from individual genes, such as scVelo, fail to accurately fit these dynamics and can not identify them as velocity genes, which further validates TSvelo's performance in complex scenarios. The visualization of dynamics fitting on more genes are provided at Fig. S8 in Supplementary Information.

From the TF-target weight matrices learned by TSvelo, we can extract key regulatory relationships. First, we calculate the mean absolute weight of each TF across its interactions with target genes. The TFs with the highest mean absolute weights are ranked and presented in Fig. 3j. Notably, KLF1, a critical TF in blood progenitors[41], is assigned the highest weight. We then examine the target genes of KLF1, highlighting those with the highest weights (Fig. 3k), among which HBA-X[42], ALAS2[43], and GYPA[44] have been previously identified as key genes in erythroid differentiation and are known to be regulated by KLF1. A time-delay pattern is also observed between KLF1 and its target genes (Fig. 3l), indicating that KLF1 expression increases first, followed by a corresponding upregulation of its target genes.

TSvelo can gene dynamics well and predict cell fate on mouse brain data

We apply TSvelo to a 10x multi-omics dataset from the embryonic mouse brain, which includes both Assay for Transposase-Accessible Chromatin with sequencing (ATAC-Seq) [45] and scRNA-seq data. This dataset was previously used in the Multivelo study, which introduced an RNA velocity model designed for multi-omics datasets to capture the dynamics between chromatin accessibility and unspliced mRNA[20]. As introduced by previous study[19], Radial Glia (RG) cells located in the outer subventricular zone give rise to neurons, astrocytes, and oligodendrocytes. During cortical development, neurons follow an inside-out layering pattern in which earlier-born neurons populate the deep cortical layers, whereas later-born neurons migrate past them to occupy more superficial layers[46]. Additionally, RG cells are capable of producing intermediate progenitor cells (IPCs), which serve as neural stem cells and generate a variety of mature excitatory neurons within the cortical layers.

Multivelo is chosen as the baseline on this study since it connects the regulation signal and RNA velocity in ATAC-unspliced-spliced space. TSvelo uses the same data processed be Multivelo for a fair comparison. The velocity stream obtained by TSvelo and Multivelo are shown in Fig. 4a and Fig. 4b, respectively. TSvelo produces velocity patterns that appear more consistent with expected differentiation trends, especially for the RG (in red) to IPCs (in brown) process. The pseudotime inferred by TSvelo is also coherent with the whole cell development process (Fig. 4c).

We next further analyze the details in the phase portrait fitting. On MEIS2 (Fig. 4d), both Multivelo and TSvelo successfully models its dynamics, which follows a clear pattern on the unspliced-spliced phase portrait. However, due to the high noise and sparsity inherent in ATAC-seq data, Multivelo encounters difficulty on genes that show significant overlap between different cell types in the unspliced–spliced phase portrait. For example, in the case of BASP1 (Fig. 4e), Multivelo incorrectly models their expression patterns as monotonically increasing, failing to capture the true dynamics, which involve initial upregulation followed by downregulation. In contrast, TSvelo's prediction consistents with known biological processes and identifies the delay from transcription to the unspliced mRNA and also the delay from unspliced to spliced mRNA (The rightmost plots in Fig. 4e). MSI2, which has been reported to be highly expressed in neural stem/progenitor cells[47], is more accurately modeled by TSvelo, exhibiting a decreasing expression pattern during the differentiation process. By comparison, Multivelo fails to capture the correct trend at the initial stage (Fig. 4f). We also show the learned transcriptional rate α as well as the predicted dynamics of unspliced and spliced RNA along pseudotime t, which clearly illustrate the time delays between transcription to unspliced and unspliced to spliced RNAs. The visualization of

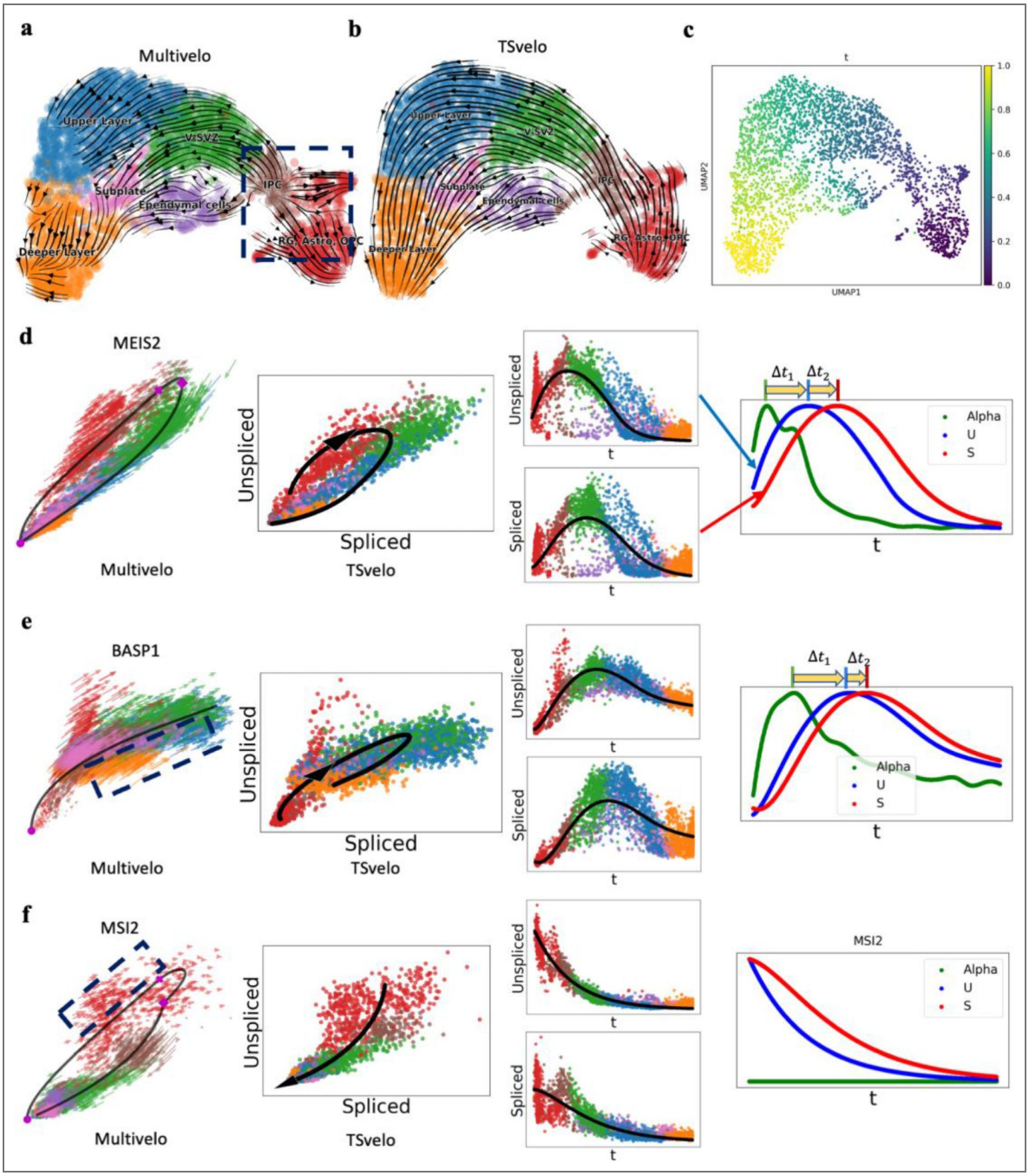


Figure 4. Results on mouse brain dataset.

(**a**) The velocity stream inferred by Multivelo. (**b**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. The pseudotime learned with TSvelo. (**d-f**) The dynamics fitting on gene MEIS2 (d), BASP1 (e) and MSI2 (f) by both Multivelo and TSvelo. In each panel, the leftmost plot shows the phase portrait fitting of Multivelo, and the next two columns show TSvelo's results. The rightmost plot shows the learned transcriptional rate (in green), unspliced abundance (in blue), and spliced abundance (in red) along the pseudotime. Since the transcriptional rate is calculated for each individual cell, we apply a Generalized Additive Model (GAM) to transcriptional representation across cells along the pseudotime and present GAM-fitting results to better visualize its trends in these plots.

dynamics fitting on more genes are provided at Fig. S9 in Supplementary Information. Given that transcriptional regulation involves the binding of TF and chromatin accessibility as measured by ATAC, TSvelo shows that it is possible to model transcriptional signals using only scRNA-Seq data.

TSvelo can predict cell fate and model lineage-specific gene dynamics for multi-lineage task

Given that multi-lineage differentiation is a common phenomenon in larger and larger scRNA-seq datasets, developing the RNA velocity model that can handle such complexity is essential. Because TSvelo demonstrates robust performance across various tasks, its applicability can be extended to more complex situations, such as scRNA-seq datasets where cells differentiate into multiple fates. We apply TSvelo to a multi-lineage dentate gyrus scRNA-seq dataset, which captures the differentiation process from neural blast cells to various cell types[48]. During preprocessing, the velocity genes selected by TSvelo are enriched in GO terms related to neural development, such as axonogenesis (GO:0007409), axon guidance (GO:0007411) and axon development (GO:0061564) (Fig. 5a), which aligns with the biological processes described by the dataset.

TSvelo could correctly identify three lineages from this data. Details about the lineage segmentation are provided at Methods and Fig. S10 in Supplementary Information. After applying the TSvelo model to each lineage and combine the results, the inferred pseudotime and velocity stream plots (Fig. 5b and Fig. 5c) align with the ground truth differentiation trajectory well, which begins with neural blast cells. We also compare TSvelo to baseline methods, including scVelo and cellDancer. Notably, cellDancer is designed to handle such multi-lineage data by modeling cell- and gene-specific transcriptional rates, splicing rates, and degradation rates using neural networks. As shown in Fig. 5d and Fig. 5e, scVelo struggles with this multi-lineage dataset, and cellDancer fails to correctly capture the trajectory of immature granule cells (colored in purple).

Next, we analyze gene expression patterns across different lineages. TSvelo provides inferred dynamics along each lineage, revealing that the expression of ANK3 in the granule and Cornu Ammonis (CA) lineages follows an increasing and then decreasing pattern. In contrast, the expression of ANK3 in the glial lineage decreases to zero (Fig. 5f). The similar pattern is also observed for other genes, such as MAP1B (Fig. 5g). Both ANK3[49] and MAP1B[50] are associated with GO terms axonogenesis and axon guidance. This observation is consistent with the fact that the granule and Cornu Ammonis[51] predominantly consist of neurons, where axons are a critical component. In contrast, glial cells, such as astrocytes, typically lack axons, providing a potential explanation for these expression patterns. SLC1A2 exhibits a distinct expression pattern, showing a significant increase in the glial lineage and a decrease in the granule and CA lineages. This pattern aligns with the observation that astrocytes are the primary cell type expressing SLC1A2[52]. Details about dynamics fitting for genes on each lineage are provided at Fig. S11 in Supplementary Information.

TSvelo can predict cell fate and model lineage-specific gene dynamics on Larry dataset

The Lineage and RNA Recovery (LARRY) method utilizes barcoded hematopoietic cells to trace both cell lineage and gene expression over time. LARRY has been successfully employed to track the in vitro differentiation of human blood cells, accurately capturing lineage trajectories and cell fates[53].

On this LARRY dataset which encompass a total of 49,302 cells on multiple lineages, TSvelo effectively detects the initial Leiden clusters based on the unspliced-to-spliced delay (Fig. 6a and Fig. 6b) and separate its lineages (Details are provided at the 'Lineage segmentation and pseudotime initialization' section in Methods). Furthermore, Fig. 6c and Fig. 6d show that the pseudotime and velocity stream inferred by TSvelo can capture the differentiation process, progressing from undifferentiated cells to distinct cell fates. The velocity genes selected by TSvelo

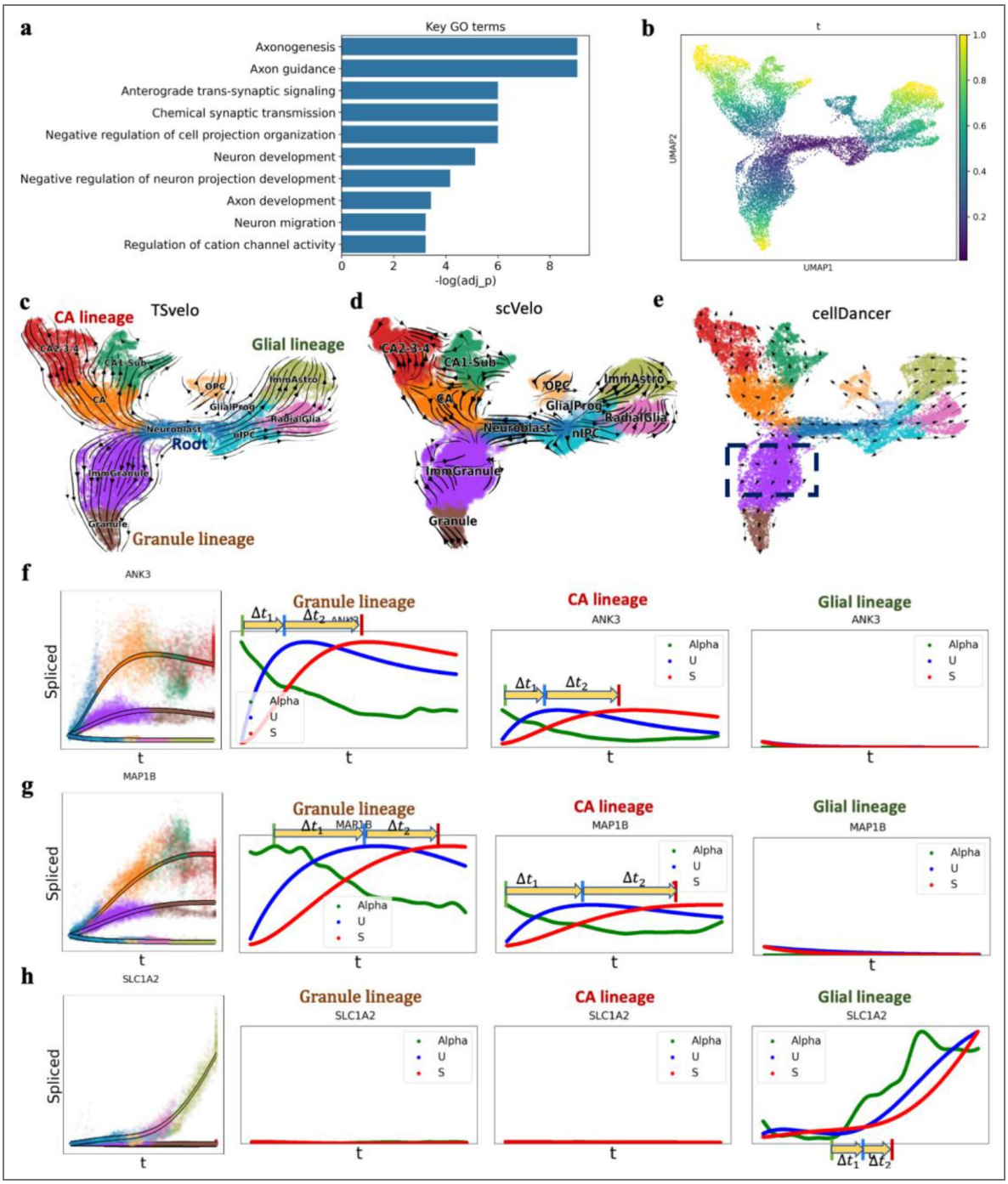


Figure 5. Results on the multi-lineage dentate gyrus dataset.

(**a**) The GO terms enriched in the selected velocity genes of TSvelo. (**b**) The pseudotime learned with TSvelo. (**c**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. Three lineages are detected, which are Granule lineage, CA lineage and glial lineage. **d** The velocity stream inferred by scVelo. (**e**) The velocity stream inferred by cellDancer.(**f-h**) The dynamics modeling of TSvelo on three axonogenesis-related genes, ANK3 (f), MAP1B (g) and SLC1A2 (h). In the leftmost plot of each panel, the lines represent the predicted spliced abundance across all lineages, with the color indicating the cell types most associated with each pseudotime point along the corresponding lineage. Additionally, expression data for each lineage are shown as translucent points. The remaining plots in each panel display the dynamics of the learned transcriptional rate (in green), unspliced abundance (in blue), and spliced abundance (in red) along pseudotime for each lineage. The transcriptional representation in these plots is also processed using GAM fitting.

are significantly enriched in processes related to neutrophil biology, such as neutrophil degranulation (GO:0043312), neutrophil activation in immune response (GO:0002283), and neutrophil-mediated immunity (GO:0002446).

We further analyzed the expression patterns of genes associated with neutrophil degranulation, focusing specifically on the neutrophil lineage. Using four representative genes as examples (Figs. 6f, g), TSvelo can help observe the significant variation in the expression patterns across these genes. PYGL, which has been widely reported as a key gene in neutrophil[54], is almost exclusively expressed in neutrophil cells and exhibits an increasing expression pattern during neutrophil differentiation. In contrast, MS4A3 was examined across all lineages, and using TSvelo, we observed that its expression initially increases and then decreases in both the neutrophil and monocyte lineages. This pattern is consistent with prior studies identifying MS4A3 as a gene specifically expressed by granulocyte-monocyte progenitors[47]. Similarly, CLEC12A, another gene associated with neutrophil degranulation, shows a comparable expression trajectory during neutrophil differentiation. Previous research has indicated that CLEC12A expression is highest in granulocyte-macrophage progenitors[55]. LTA4H exhibits a pattern similar to that of CLEC12A, suggesting it may play a significant role in the early stages of neutrophil differentiation. These findings further confirm the utility of TSvelo for gene-level analysis in multi-lineage differentiation datasets.

Discussion

RNA velocity, utilizing unspliced/spliced data, has become a widely adopted concept for predicting cell fate and modeling gene dynamics. While several RNA velocity models have been proposed, most of they are based on phase portrait fitting in the unspliced-spliced space. However, due to the limited information and high noise inherent in the splicing data of individual genes, the short time delays between unspliced and spliced abundance, and the challenges posed by large-scale datasets with complex processes, most genes cannot be accurately modeled by previous RNA velocity methods. This limitation reduces the reliability and robustness of downstream analyses.

Gene expression is a complex biological process within cells, involving multiple regulatory mechanisms from DNA to the matured RNA. In this process, transcription and splicing are two crucial steps to determine the final gene expression. Due to the mathematical complexity, previous methods cannot jointly model gene regulation, transcription and splicing. TSvelo comprehensively models the cascade of the whole process using an interpretable ODE framework, allowing for learning dynamics of all velocity genes simultaneously. TSvelo could accurately model gene dynamics, predict cell fate, detect the key regulatory relations, and handle multi-lineage datasets. Results on six scRNA-seq datasets demonstrate that TSvelo is a valuable approach for RNA velocity modeling (results on pons dataset are provided in Fig. S3 in Supplementary Information).

One limitation of TSvelo is the reliance on predefined TF–target regulatory priors, which may be incomplete and may not fully capture context-specific regulatory relationships in primary tissues or at single-cell resolution. In addition, modeling both regulatory interactions across high-dimensional genes and RNA kinetics introduces extra computational overhead relative to some existing approaches (runtime and memory analyses are provided in Table R2, Table R3 and Fig. S16).

Looking forward, there are several opportunities to further strengthen the framework. Incorporating context-specific regulatory information, such as single-cell chromatin accessibility data, may improve transcriptional modeling. Moreover, TSvelo currently assumes a constant gene-specific splicing and degradation rates. Since splicing factors also play significant roles in regulating other genes during the splicing process[56], and there are still complex mechanisms of mRNA degradation[57], further exploration could enhance velocity's modelling and bring new biological insights in future.

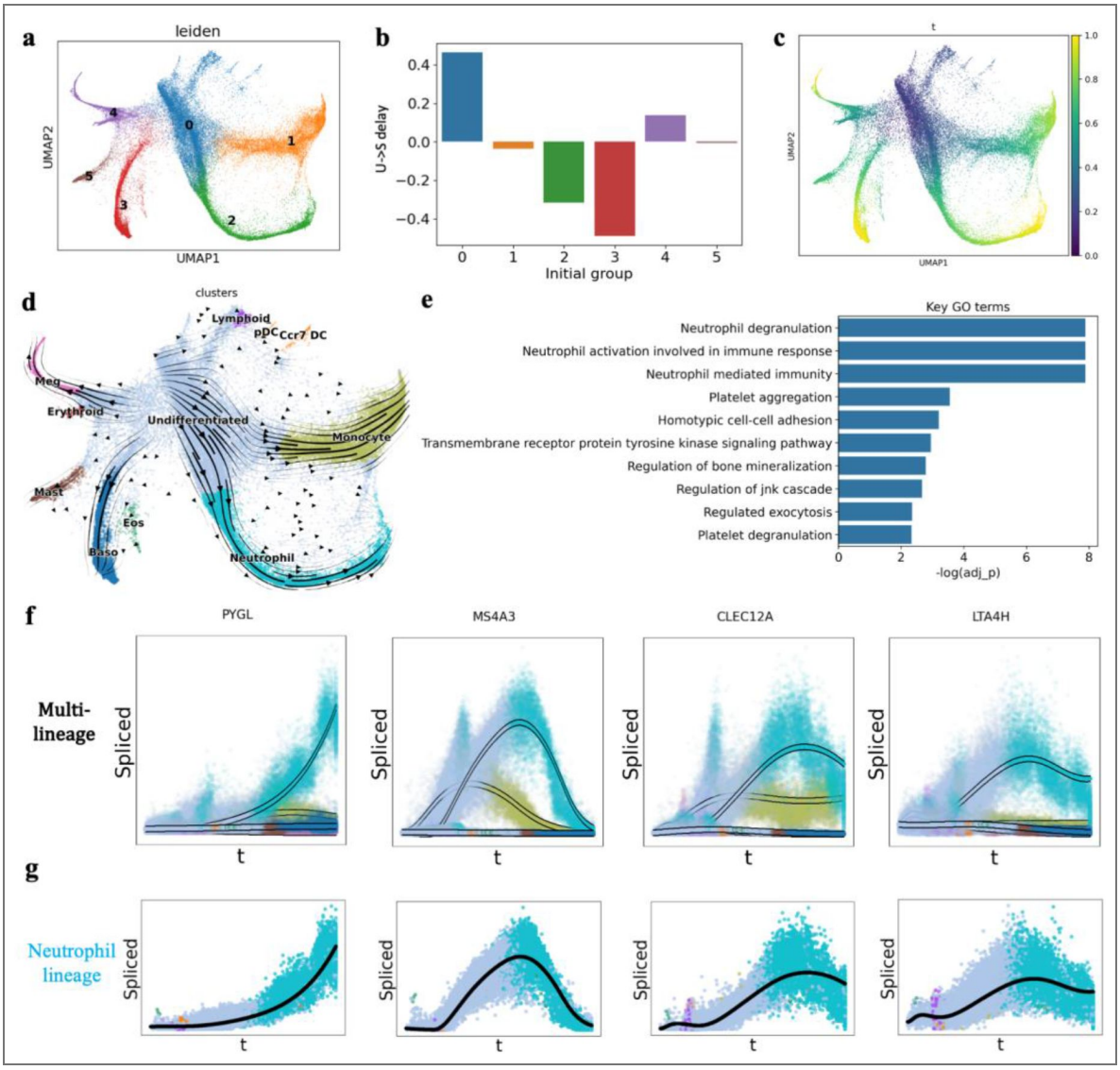


Fig 6. Results on the LARRY dataset.

(**a**) The Leiden clustering on LARRY. (**b**) The initial Leiden cluster detection in preprocessing. (**c**) The pseudotime learned with TSvelo. (**d**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. (**e**) The GO terms enriched in the selected velocity genes of TSvelo. (**f**) The dynamics modeling of TSvelo on four genes related to neutrophil development, which are PYGL, MS4A3, CLEC12A and LTA4H. The lines represent the predicted spliced abundance across all lineages, with the color indicating the cell types most strongly associated with each pseudotime point along the corresponding lineage. **(g)** The dynamics modeling of PYGL, MS4A3, CLEC12A and LTA4H on the neutrophil lineage.

Methods

Preprocessing for scRNA-seq data

The unspliced and spliced RNA abundances are preprocessed with multiple steps, which includes highly variable genes (HVGs) selection, normalization, log transformation, k-nearest neighbor (KNN) smoothing, and clustering. Following the data preprocessing procedures outlined in scVelo[6], we first select the top 2,000 highly variable genes (HVGs) and normalize their expression profiles by dividing by the total counts in each cell. A nearest-neighbor graph (with 30 neighbors by default) was constructed based on Euclidean distances in principal component analysis space (with 30 principal components by default) on log-transformed gene expression data. Subsequently, we compute the first- and second-order moments (mean and uncentered variance) for each cell across its nearest neighbors. These steps are performed by using scvelo.pp.filter_and_normalize() and scvelo.pp.moments().

Acquiring Prior Knowledge of Gene Regulatory Relations

To construct a prior gene network for the selected HVGs, we utilize TF-target annotations from the ENCODE[58] and ChEA[59] databases. If a regulatory interaction between a transcription factor (TF) and its target gene is identified in either of these two databases, the TF is included in the set of regulators for modeling the dynamic behavior of the target gene. More analysis on the selection of TF-target database is shown in Fig. S13 and Fig. S14 in Supplementary Information.

Velocity genes selection

Previous studies[6] have demonstrated that only a subset of genes, termed "velocity genes," can be accurately fitted in the unspliced-spliced phase portrait by RNA velocity models. Inspired by the notion that velocity genes provide high-quality data for dynamic modeling in the phase portrait, we propose to select velocity genes during the preprocessing step. The velocity genes are selected based on the premise that they exhibit clear dynamics in the unspliced-spliced phase portrait, which is necessary for precise modeling splicing dynamics. The selection is based on the similarity between cell-cell neighborhood relationships in UMAP space and those observed in gene-specific unspliced-spliced phase portraits. In detail, we first compute the cell-cell neighborhood graph, *Graph*, using scvelo.pp.neighbors() on all highly variable genes (HVGs). Next, we construct an anndata object $adata_g$ for each gene, which contains only the unspliced and spliced expression data of that gene. We then apply scvelo.pp.neighbors() to each $adata_g$ to calculate the gene-specific neighborhood graph, denoted as $Graph_g$. Subsequently, we compute the similarity between *Graph* and each $Graph_g$. The top 100 genes with the highest similarity are selected as velocity genes. These genes are characterized by phase portraits that exhibit a structure similar to that of the UMAP space, thereby enhancing the separation of cells from different types. As a result, the splicing dynamics of these velocity genes are more likely to be captured by RNA velocity models.

lineages segmentation and pseudotime initialization

Based on the normalized gene expression data, we perform Leiden clustering using scanpy.tl.leiden() with a low resolution (default resolution = 0.1). These Leiden clusters are utilized to identify lineages and determine the differentiation direction for each lineage. Subsequently, we apply PAGA (scanpy.tl.paga()) to assess the connectivity between Leiden groups.

By setting one Leiden group as the initial state, we infer pseudotime using diffusion pseudotime (DPT) with scanpy.tl.dpt(). To detect lineages, we start from the selected initial Leiden cluster and filter the PAGA graph by applying a threshold (default = 0.02). Edges with weights below this threshold are removed, leaving only the shortest path from the initial state to each group. The remaining paths are considered as the detected lineages.

Given a lineage, TSvelo could detect the orientation of differentiation along it, which is based on the fact that unspliced RNA precedes spliced RNA in the expression pattern[60]. For each gene, TSvelo computes the Spearman correlation between unspliced and spliced abundance across

different moving steps along the DPT. The time number of moving steps along unspliced to spliced expression, which maximizes the Spearman correlation, is considered the U-to-S delay (Fig. 1a shows an example of U-to-S delay with gene EML5). The U-to-S delay for each lineage is calculated as the mean U-to-S delay for all genes on it. And the average U-to-S delay across all lineages provides the overall U-to-S delay for the initial Leiden cluster. Finally, the initial Leiden cluster with the highest U-to-S delay is considered the correct initiation. The DPT under this condition is used to initialize pseudotime for TSvelo. If multiple lineages are detected, TSvelo will model each lineage independently and subsequently merge them in the final step. A detailed explanation with example illustrating how lineages are segmented is provided in the Fig. S10 in Supplementary Information. The initialization results on all datasets are shown in Fig. S12. We also provide a simulation study to verify this initialization strategy in the Fig. S2.

The ODE model in TSvelo

Suppose u_g and s_g are the abundance of unspliced and spliced RNA, and $\alpha_g(t)$, β_g, γ_g are the transcription rate, splicing rate and degradation rate, respectively. The dynamics of a velocity gene g is modeled as:

$$\frac{du_g(t)}{dt} = \alpha_g(t) - \beta_g u_g(t) \tag{4}$$

$$\frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t) \tag{5}$$

We assume that the gene and cell-specific transcriptional rate $\alpha_g(t)$ is influence by the expression of Transcriptional Factors (TFs), while β_g and γ_g are gene-specific constant parameters. Considering the wide usage of linear models for gene-relations in previous studies[27,34-37], we model $\alpha_g(t)$ as

$$\alpha_g(t) = \mathrm{ReLU}\left(\sum_{i \in TFS(g)} w_{gi} * s_i(t)\right) \tag{6}$$

In order to include more TFs in the modeling, we reserve those TFs that are not selected as velocity genes. These TFs are excluded from the velocity genes selection because their unspliced-spliced expression does not provide sufficient information, primarily due to high noise in the unspliced RNA. Consequently, we incorporate these TFs, whose spliced abundance is denoted as s', into the TSvelo by directly modeling the process between transcriptional signal to mature RNA,

$$\frac{ds_g^{'}(t)}{dt} = \alpha_g(t) - \gamma_g s_g^{'}(t) \tag{7}$$

As a result, we can get the ODE model with parameters matrix A:

$$\begin{bmatrix} dU/dt \\ dS/dt \\ dS^{'}/dt \end{bmatrix} = A \begin{bmatrix} U \\ S \\ S^{'} \end{bmatrix} = \begin{bmatrix} -B & W & W^{'} \\ B & -\Gamma & 0 \\ 0 & W^{'} & W^{'} - \Gamma^{'} \end{bmatrix} \begin{bmatrix} U \\ S \\ S^{'} \end{bmatrix} \tag{8}$$

$U = [u_1, u_2 \ldots, u_G]^T \in R^G$, $S = [s_1, s_2 \ldots, s_G]^T \in R^G$, $S' = [s_{G+1}, s_{G+2} \ldots, s_{G+K}]^T \in R^K$, G is the number of velocity genes and K is the number of additional TFs which are not selected as velocity genes. $B \in R^{G \times G}$ and $\Gamma \in R^{G \times G}$ are the diagonal matrices consist of splicing rates [β_1,

β_2 ..., β_G] and degradation rates [γ_1, γ_2 ..., γ_G] for velocity genes, respectively. $\Gamma' \in R^{K\times K}$ is the diagonal matrices consist of degradation rates [γ_{G+1}, γ_{G+2} ..., γ_{G+K}] for these additional TFs. $W \in R^{G\times G}$ and $W' \in R^{K\times K}$ denote the matrices representing the TF-target relationships for the TFs selected as velocity genes and those not selected as velocity genes, respectively. Using these gene-gene weight matrices, TSvelo can model the gene- and cell-specific transcriptional rate α_g. The whole parameters matrix $A \in R^{(2G+K)\times(2G+K)}$

Optimizing global time and Neural ODE in EM framework

In TSvelo, the dynamics described in Eq. 7 are implemented using a neural ordinary differential equation (ODE) model. Given an initial state and the parameters in matrix A, the neural ODE model can compute the values of U and S at any time step. This approach eliminates the need for an analytical solution for unspliced and spliced expression, enhancing the flexibility of TSvelo. By default, the number of time steps in the ODE is set to 1,000. The ReLU activation function is applied to the parameters α_g, β_g and γ_g, in order to prevent them from taking negative values. Notably, TSvelo does not incorporate deep neural networks or encoders. Since TSvelo models all genes simultaneously, we normalize the unspliced and spliced abundances of each gene by its standard deviation before inputting the data into the Neural ODE model to ensure that the influence of each gene is balanced.

TSvelo optimizes the pseudotime t and parameters matrix A in ODE iteratively using an Expectation-Maximization (EM) approach. The maximum number of iterations is set to 30 by default, with an early stopping criterion applied. Given the parameters in ODE system, we can compute the predicted values of unspliced (for velocity genes) and spliced (for velocity genes and additional TFs) at all time steps, denoted as $; A)$ and $; A)$, where $t_{\text{step}} \in$ {0,1,2 ...,999}. Given the time steps assignment t_c for each cell c, we can get the expected values for velocity genes at all cells as $\widehat{U}(t_c; A)$, $\hat{S}(t_c; A)$ and $\widehat{S}'(t_c; A)$. The loss function used in EM algorithm is the mean squared error between the expected value with the observed data.

$$\text{Loss} = \frac{1}{c}\sum_c \left[\left(\widehat{U}(t_c; A) - U_c\right)^2 + \left(\hat{S}(t_c; A) - S_c\right)^2 + \left(\widehat{S}'(t_c; A) - S_c'\right)^2\right] \quad (9)$$

The EM algorithm will iteratively update the time assignment and parameters in neural ODE by minimizing the above loss function.

In the E-step, following the strategy adopted in scVelo model[6], TSvelo updates the time step assigned to each cell using grid search with time steps range $t_{step} \in$ {0,1,2 ...,999}, which is achieved by finding the minimum distance between the observed unspliced-spliced expression and the model prediction ($\widehat{U}$ and $\hat{S}$) at all time steps.

$$\min_{t_c} \frac{1}{c}\sum_c \left[\left(\widehat{U}(t_c; A) - U_c\right)^2 + \left(\hat{S}(t_c; A) - S_c\right)^2 + \left(\widehat{S}'(t_c; A) - S_c'\right)^2\right] \quad (10)$$

In the M-step, the parameter matrix A is updated using gradient descent within the neural ODE model to minimize the loss function. Here we use the prior gene regulation knowledge to constraint the weight matrix W and W' in A. Only If gene i is annotated as a TF for gene j in the ENCODE or ChEA databases, the corresponding w_{ij} could be learnable. Otherwise w_{ij} is fixed at zero. The usage of prior knowledge could keep the matrix sparse and avoid overfitting. The neural ODE module is implemented using the torchdiffeq package and trained with the Adam optimizer, a learning rate of 0.05, and a maximum of 500 epochs, incorporating an early stopping criterion to prevent overfitting.

$$\min_{A} \frac{1}{c} \sum_{c} \left[\left(\widehat{U}\left(t_c; A\right) - U_c \right)^2 + \left(\hat{S}\left(t_c; A\right) - S_c \right)^2 + \left(\widehat{S}^{'}\left(t_c; A\right) - S_c^{'} \right)^2 \right] \tag{11}$$

By performing the EM optimization, TSvelo could fit the high-dimensional unspliced-spliced data across multiple genes well, and learn the global pseudotime for each cell. After the training procedure, the RNA velocity could be calculated using those parameters in matrix A, which could be further used for cell fate prediction.

Metrics for evaluating

1. Velocity consistency (VCon). We used the scvelo.velocity_confidence() function from scVelo to evaluate velocity consistency, interpreting the results as a measure of how consistent velocities are within neighboring cells. Velocity consistency is especially suitable for evaluating the RNA velocity modeling on single lineage. For each cell *c*, the velocity consistency is calculated as follows:

$$\mathrm{VCon}(c) = \frac{1}{\left|\{c^{'} \in N(c)\}\right|} \sum_{c^{'} \in N(c)} \frac{\boldsymbol{v}_c \cdot \boldsymbol{v}_{c^{'}}}{|\boldsymbol{v}_c| \cdot |\boldsymbol{v}_{c^{'}}|} \tag{12}$$

   where $N(c)$ represents the neighboring cells of a given cell *c*. $\boldsymbol{v}_c$ and $\boldsymbol{v}_{c^{'}}$ denote the low-dimensional velocity vectors of cell *c* and its neighboring cell *c*'.
2. Cross-boundary direction correctness (CBDir). Cross-boundary direction correctness is initially introduced by VeloAE[61], which assesses the accuracy of transitions from a source cluster to a target cluster by examining the boundary cells, and requires ground truth annotations. We directly run the function unitvelo.evaluate() provided in UniTVelo[7] to obtain the Cross-boundary direction correctness. In detail, the CBDir is calculated as follows:

$$\mathrm{CBDir}(c) = \frac{1}{\left|\{c^{'} \in C_A \cap N(c)\}\right|} \sum_{c^{'} \in C_A \cap N(c)} \frac{\boldsymbol{v}_c \cdot (\boldsymbol{x}_{c^{'}} - \boldsymbol{x}_c)}{|\boldsymbol{v}_c| \cdot |\boldsymbol{x}_{c^{'}} - \boldsymbol{x}_c|} \tag{13}$$

   where C_A denotes the set of cells in the target cluster A, and $N(c)$ represents the neighboring cells of a given cell *c*. $\boldsymbol{v}_c$ and $\boldsymbol{x}_c$ denote the low-dimensional velocity and state vectors of cell *c*, respectively, and $\boldsymbol{x}_{c^{'}}$ denotes the state vector of its neighboring cell.
3. Within-cluster velocity coherence (ICCoh). Within-cluster velocity coherence is initially introduced by VeloAE[61], which measures the coherence of velocities within a single cluster using a cosine similarity score between cell velocities. We applied the function unitvelo.evaluate() provided by UniTVelo[7] to directly compute the within-cluster velocity coherence. Using the same notation as defined above, the CBDir is calculated as follows:

$$\mathrm{ICCoh}(c) = \frac{1}{\left|\{c^{'} \in C_A \cap N(c)\}\right|} \sum_{c^{'} \in C_A \cap N(c)} \frac{\boldsymbol{v}_c \cdot \boldsymbol{v}_{c^{'}}}{|\boldsymbol{v}_c| \cdot |\boldsymbol{v}_{c^{'}}|} \tag{14}$$

The comparison between the 3D phase portrait and traditional 2D phase portrait

To quantitatively assess whether TSvelo can distinguish cell types, we evaluated the separability of cell-type labels in both the 2D (unspliced–spliced) phase portrait adopted by previous RNA velocity approaches, and the 3D (α–unspliced–spliced, α denotes the transcriptional rate) phase portrait introduced by TSvelo.

Specifically, we evaluated how well the embedding preserves cell-type information using a k-nearest neighbors (kNN) classification accuracy with 5-fold cross-validation. Given an embedding matrix in 2D or 3D space ($X \in \mathbb{R}^{n*d}$, where n is the number of cells and d is 2 or 3) and

corresponding cell-type labels ($y \in \{1, \ldots, C\}$), we partition the data into five folds. For each fold (k), a kNN classifier with $K = 5$, denoted as $f^{(k)}$, is trained on the training subset and evaluated on the held-out test subset. The classification accuracy for the k-th fold is defined as

$$Acc^{(k)} = \frac{1}{n_k} \sum_{i \in D_k^{\text{test}}} \mathbf{1}\left(f^{(k)}(x_i) = y_i\right) \tag{15}$$

where n_k is the number of samples in the test set and $\mathbf{1}(\cdot)$ is the indicator function. The final score is obtained by averaging across all folds:

$$Acc = \frac{1}{5} \sum_{k=1}^{5} Acc^{(k)} \tag{16}$$

This metric directly assesses whether cells of the same type are positioned close to each other in the embedding space, and is widely used to quantify representation quality.

GO term enrichment analysis

Gene Ontology (GO) terms enrichment analysis is a widely used method for identifying biological processes, molecular functions, or cellular components that are overrepresented in a given set of genes compared to a background set. Here we use the biological processes enrichment analysis in GO terms database to explore the associated biological roles of those selected velocity genes. To perform the GO term enrichment analysis, we utilized the “gseapy.enrichr()” function in Python with using Fisher’s exact test by default.

Merging of lineage-specific results

For datasets containing multiple lineages, TSvelo applies its model independently to each lineage. The resulting lineage-specific outputs are integrated into a unified representation by aggregating both expression layers and cell-level annotations across branches.

All objects are aligned to a common gene space by restricting to the shared set of genes. Let $b = 1, \ldots, B$ index the lineage-specific results. For each cell c and each inferred quantity (e.g. velocity and psuedotime), TSvelo collects the corresponding results $x_{c,b}$ from all lineages in which the cell is present. Layer values are merged on a per-cell basis using a weighted average, where the weight for each branch is given by its number of cells, n_b. The merged value for cell c is computed as

$$\tilde{x}_c = \frac{\sum_b n_b \cdot x_{c,b}}{\sum_b n_b} \tag{17}$$

Continuous variables are averaged, while discrete variables are cast to integer values after aggregation.

Finally, the aggregated layer values and annotations are assigned back according to the global cell index, producing a unified dataset that integrates lineage-specific inferences.

Data availability

The pancreatic endocrinogenesis dataset comprises the single-cell RNA-seq (10X) data of pancreatic epithelial and Ngn3-Venus fusion cells sampled from mouse embryonic day 15.5, which could be loaded using scVelo's package scvelo.datasets.pancreas(). The gastrulation erythroid dataset, which is selected from the transcriptional profiles of mouse embryos39, which could be loaded using scVelo's package scvelo.datasets.pancreas(). 10x embryonic mouse brain dataset is provided at the 10x website at *https://www.10xgenomics.com/resources/datasets/fresh-embryonic-e-18-mouse-brain-5-k-1-standard-1-0-0*. The data preprocessed by Multivelo is utilized in this study, (*https://multivelo.readthedocs.io/en/latest/MultiVelo_Fig2.html*). The dentate gyrus neurogenesis

data is available at *http://pklab.med.harvard.edu/velocyto/DentateGyrus/DentateGyrus.loom* The LARRY dataset has been shared by pyrovelocity, which could be accessed at *https://figshare.com/articles/dataset/larry_invitro_adata_sub_raw_h5ad/20780344*. The raw data of Hindbrain (pons) of adolescent mice is from *https://pklab.med.harvard.edu/ruslan/velocity/oligos/*. The ENCODE TF-target database website: *https://maayanlab.cloud/Harmonizome/dataset/ENCODE+Transcription+Factor+Targets*. The ChEA TF-target database website: *https://maayanlab.cloud/Harmonizome/dataset/CHEA+Transcription+Factor+Targets*. The results of BayVel on the pancreas dataset is downloaded from its GitHub page at *https://github.com/elenasabbioni/BayVel_notebooks/tree/main/real%20data/Pancreas/moments/output*. TSvelo is implemented in Python. The source code can be downloaded from the GitHub repository, *https://github.com/lijc0804/TSvelo*.

Acknowledgements

This work was supported by the National Key R&D Program of China (2023YFF1204500 to Y.Y.), and the National Natural Science Foundation of China (No. 62503452 to J.L.).

Additional files

Supplementary material.

Additional information

Funding

Funder	Grant reference number	Author
MOST | National Key Research and Development Program of China (NKPs)	2023YFF1204500	Ye Yuan
MOST | National Natural Science Foundation of China (NSFC)	62503452	Jiachen Li

Author ORCID iDs

Jiachen Li: https://orcid.org/0000-0001-9137-6262
Ye Yuan: https://orcid.org/0009-0003-7969-469X

References

1 **Wolf F. A.,** et al. (2019) PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome biology* **20**:1-9 https://doi.org/10.1186/s13059-019-1663-x | PubMed

2 **Qiu X.,** et al. (2017) Reversed graph embedding resolves complex single-cell trajectories. *Nature methods* **14**:979-982 https://doi.org/10.1038/nmeth.4402 | PubMed

3 **Street K.,** et al. (2018) Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC genomics* **19**:1-16 https://doi.org/10.1186/s12864-018-4772-0 | PubMed

4 **Setty M.,** et al. (2019) Characterization of cell fate probabilities in single-cell data with Palantir. *Nature biotechnology* **37**:451-460 https://doi.org/10.1038/s41587-019-0068-4 | PubMed

5 **La Manno G.,** et al. (2018) RNA velocity of single cells. *Nature* **560**:494-498 https://doi.org/10.1038/s41586-018-0414-6 | PubMed

6 **Bergen V.,** Lange M., Peidli S., Wolf F. A., Theis F. J. (2020) Generalizing RNA velocity to transient cell states through dynamical modeling. *Nature biotechnology* **38**:1408-1414 https://doi.org/10.1038/s41587-020-0591-3 | PubMed

7 **Gao M.,** Qiao C., Huang Y. (2022) UniTVelo: temporally unified RNA velocity reinforces single-cell trajectory inference. *Nature Communications* **13**:6586 https://doi.org/10.1038/s41467-022-34188-7 | PubMed

8 **Gayoso A.,** et al. (2024) Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. *Nature methods* **21**:50-59 https://doi.org/10.1038/s41592-023-01994-w | PubMed

9 **Gu Y.,** Blaauw D. T., Welch J. (no date) International Conference on Machine Learning. PMLR. pp. 7887-7901

10 **Qin Q.,** Bingham E., La Manno G., Langenau D. M., Pinello L. (2022) Pyro-Velocity: Probabilistic RNA Velocity inference from single-cell data. *bioRxiv* https://doi.org/10.1101/2022.09.12.507691

11 **Sabbioni E.,** Bibbona E., Mastrantonio G., Sanguinetti G. (2025) BayVel: A Bayesian Framework for RNA Velocity Estimation in Single-Cell Transcriptomics. *arXiv* https://doi.org/10.48550/arxiv.2505.03083

12 **Li S.,** et al. (2023) A relay velocity model infers cell-dependent RNA velocity. *Nature Biotechnology* **42**:99-108 https://doi.org/10.1038/s41587-023-01728-5 | PubMed

13 **Cui H.,** et al. (2024) DeepVelo: deep learning extends RNA velocity to multi-lineage systems with cell-specific kinetics. *Genome Biology* **25**:27 https://doi.org/10.1186/s13059-023-03148-9 | PubMed

14 **Farrell S.,** Mani M., Goyal S. (2023) Inferring single-cell transcriptomic dynamics with structured latent gene expression dynamics. *Cell Reports Methods* **3** https://doi.org/10.1016/j.crmeth.2023.100581 | PubMed

15 **Qiu X.,** et al. (2022) Mapping transcriptomic vector fields of single cells. *Cell* **185**:690-711.e645 https://doi.org/10.1016/j.cell.2021.12.045 | PubMed

16 **Ma A.,** McDermaid A., Xu J., Chang Y., Ma Q. (2020) Integrative methods and practical challenges for single-cell multi-omics. *Trends in biotechnology* **38**:1007-1022 https://doi.org/10.1016/j.tibtech.2020.02.013 | PubMed

17 **Subramanian I.,** Verma S., Kumar S., Jere A., Anamika K. (2020) Multi-omics data integration, interpretation, and its application. *Bioinformatics and biology insights* **14**:1177932219899051 https://doi.org/10.1177/1177932219899051 | PubMed

18 **Gorin G.,** Svensson V., Pachter L. (2020) Protein velocity and acceleration from single-cell multiomics experiments. *Genome biology* **21**:1-6 https://doi.org/10.1186/s13059-020-1945-3 | PubMed

19 **Li C.,** Virgilio M. C., Collins K. L., Welch J. D. (2023) Multi-omic single-cell velocity models epigenome–transcriptome interactions and improves cell fate prediction. *Nature Biotechnology* **41**:387-398 https://doi.org/10.1038/s41587-022-01476-y | PubMed

20 **Riba A.,** et al. (2022) Cell cycle gene regulation dynamics revealed by RNA velocity and deep-learning. *Nature communications* **13**:1-13 https://doi.org/10.1038/s41467-022-30545-8 | PubMed

21 **Lederer A. R.,** et al. (2024) Statistical inference with a manifold-constrained RNA velocity model uncovers cell cycle speed modulations. *Nature Methods* **21**:2271-2286 https://doi.org/10.1038/s41592-024-02471-8 | PubMed

22 **Zhou P.,** Bocci F., Li T., Nie Q. (2024) Spatial transition tensor of single cells. *Nature Methods* **21**:1053-1062 https://doi.org/10.1038/s41592-024-02266-x | PubMed

23 **Abdelaal T.,** et al. (2024) SIRV: Spatial inference of RNA velocity at the single-cell resolution. *NAR genomics and bioinformatics* **6**:qae100 https://doi.org/10.1093/nargab/lqae100 | PubMed

24 **Liu R.,** Pisco A. O., Braun E., Linnarsson S., Zou J. (2022) Dynamical Systems Model of RNA Velocity Improves Inference of Single-cell Trajectory, Pseudo-time and Gene Regulation. *Journal of Molecular Biology* **167606** https://doi.org/10.1016/j.jmb.2022.167606 | PubMed

25 **Gorin G.,** Fang M., Chari T., Pachter L. (2022) RNA velocity unraveled. *PLOS Computational Biology* **18**:e1010492 https://doi.org/10.1371/journal.pcbi.1010492 | PubMed

26 **Soneson C.,** Srivastava A., Patro R., Stadler M. B. (2021) Preprocessing choices affect RNA velocity results for droplet scRNA-seq data. *PLoS computational biology* **17**:e1008585 https://doi.org/10.1371/journal.pcbi.1008585 | PubMed

27 **Li J.,** Pan X., Yuan Y., Shen H.-B. (2024) TFvelo: gene regulation inspired RNA velocity estimation. *Nature Communications* **15**:1387 https://doi.org/10.1038/s41467-024-45661-w | PubMed

28 **Bergen V.,** Soldatov R. A., Kharchenko P. V., Theis F. J. (2021) RNA velocity—current challenges and future perspectives. *Molecular systems biology* **17**:e10282 https://doi.org/10.15252/msb.202110282 | PubMed

29 **Hossain I.,** Fanfani V., Fischer J., Quackenbush J., Burkholz R. (2024) Biologically informed NeuralODEs for genome-wide regulatory dynamics. *Genome Biology* **25**:127 https://doi.org/10.1186/s13059-024-03264-0 | PubMed

30 **Burdziak C.,** et al. (2023) scKINETICS: inference of regulatory velocity with single-cell transcriptomics data. *Bioinformatics* **39**:i394-i403 https://doi.org/10.1093/bioinformatics/btad267 | PubMed

31 **Zhou Z.,** Li J., Xin H., Pan X., Shen H.-B. (2025) Simultaneously infer cell pseudotime, velocity field, and gene interaction from multi-branch scRNA-seq data with scPN. *NAR Genomics and Bioinformatics* **7**:qaf144 https://doi.org/10.1093/nargab/lqaf144 | PubMed

32 **Chen R. T.,** Rubanova Y., Bettencourt J., Duvenaud D. K. (2018) Neural ordinary differential equations. In: Advances in Neural Information Processing Systems 31 (NeurIPS 2018).

33 **Gao M.,** et al. (2025) CLADES: a hybrid NeuralODE-Gillespie approach for unveiling clonal cell fate and differentiation dynamics. *Nature Communications* **16**:8174 https://doi.org/10.1038/s41467-025-63150-6 | PubMed

34 **Tran A.,** Yang P., Yang J. Y., Ormerod J. T. (2022) scREMOTE: Using multimodal single cell data to predict regulatory gene relationships and to build a computational cell reprogramming model. *NAR genomics and bioinformatics* **4**:qac023 https://doi.org/10.1093/nargab/lqac023 | PubMed

35 **Song Q.,** Ruffalo M., Bar-Joseph Z. (2023) Using single cell atlas data to reconstruct regulatory networks. *Nucleic Acids Research* **51**:e38-e38 https://doi.org/10.1093/nar/gkad053 | PubMed

36 **Liu Y.-Y.,** Slotine J.-J., Barabási A.-L. (2011) Controllability of complex networks. *nature* **473**:167-173 https://doi.org/10.1038/nature10011 | PubMed

37 **Wang L.,** et al. (2023) Dictys: dynamic gene regulatory network dissects developmental continuum with single-cell multiomics. *Nature Methods* **20**:1368-1378 https://doi.org/10.1038/s41592-023-01971-3 | PubMed

38 **Weiler P.,** Van den Berge K., Street K., Tiberi S. (2022) *Single Cell Transcriptomics: Methods and Protocols* Springer. pp. 269-292

39 **Pijuan-Sala B.,** et al. (2019) A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature* **566**:490-495 https://doi.org/10.1038/s41586-019-0933-9 | PubMed

40 **Liu Y.,** Karlsson S. (2024) Perspectives of current understanding and therapeutics of Diamond-Blackfan anemia. *Leukemia* **38**:1-9 https://doi.org/10.1038/s41375-023-02082-w | PubMed

41 **Mukherjee K.,** Bieker J. J. (2022) EKLF/Klf1 regulates erythroid transcription by its pioneering activity and selective control of RNA Pol II pause-release. *Cell reports* **41** https://doi.org/10.1016/j.celrep.2022.111830 | PubMed

42 **Perseu L.,** et al. (2011) KLF1 gene mutations cause borderline HbA2. *Blood, The Journal of the American Society of Hematology* **118**:4454-4458 https://doi.org/10.1182/blood-2011-04-345736 | PubMed

43 **Singleton B. K.,** Burton N. M., Green C., Brady R. L., Anstee D. J. (2008) Mutations in EKLF/KLF1 form the molecular basis of the rare blood group In (Lu) phenotype. *Blood, The Journal of the American Society of Hematology* **112**:2081-2088 https://doi.org/10.1182/blood-2008-03-145672 | PubMed

44 **Kuvardina O. N.,** et al. (2015) RUNX1 represses the erythroid gene expression program during megakaryocytic differentiation. *Blood, The Journal of the American Society of Hematology* **125**:3570-3579 https://doi.org/10.1182/blood-2014-11-610519 | PubMed

45 **Buenrostro J. D.,** Giresi P. G., Zaba L. C., Chang H. Y., Greenleaf W. J. (2013) Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nature methods* **10**:1213-1218 https://doi.org/10.1038/nmeth.2688 | PubMed

46 **Nadarajah B.,** Parnavelas J. G. (2002) Modes of neuronal migration in the developing cerebral cortex. *Nature Reviews Neuroscience* **3**:423-432 https://doi.org/10.1038/nrn845 | PubMed

47 **Liu Z.,** et al. (2019) Fate mapping via Ms4a3-expression history traces monocyte-derived cells. *Cell* **178**:1509-1525.e1519 https://doi.org/10.1016/j.cell.2019.08.009 | PubMed

48 **Hochgerner H.,** Zeisel A., Lönnerberg P., Linnarsson S. (2018) Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell RNA sequencing. *Nature neuroscience* **21**:290-299 https://doi.org/10.1038/s41593-017-0056-2 | PubMed

49 **Leussis M. P.,** Madison J. M., Petryshen T. L. (2012) Ankyrin 3: genetic association with bipolar disorder and relevance to disease pathophysiology. *Biology of mood & anxiety disorders* **2**:1-13 https://doi.org/10.1186/2045-5380-2-18 | PubMed

50 **Meixner A.,** et al. (2000) MAP1B is required for axon guidance and is involved in the development of the central and peripheral nervous system. *The Journal of cell biology* **151**:1169-1178 https://doi.org/10.1083/jcb.151.6.1169 | PubMed

51 **Douglas R. J.** (1967) The hippocampus and behavior. *Psychological bulletin* **67**:416 https://doi.org/10.1037/h0024599 | PubMed

52 **Sun Y.,** et al. (2023) Promotion of astrocyte-neuron glutamate-glutamine shuttle by SCFA contributes to the alleviation of Alzheimer's disease. *Redox Biology* **62**:102690 https://doi.org/10.1016/j.redox.2023.102690 | PubMed

53 **Weinreb C.,** Rodriguez-Fraticelli A., Camargo F. D., Klein A. M. (2020) Lineage tracing on transcriptional landscapes links state to fate during differentiation. *Science* **367**:eaaw3381 https://doi.org/10.1126/science.aaw3381 | PubMed

54 **Borella R.,** et al. (2022) Metabolic reprograming shapes neutrophil functions in severe COVID-19. *European Journal of Immunology* **52**:484-502 https://doi.org/10.1002/eji.202149481 | PubMed

55 **Bill M.,** et al. (2018) Mapping the CLEC 12A expression on myeloid progenitors in normal bone marrow; implications for understanding CLEC 12A-related cancer stem cell biology. *Journal of Cellular and Molecular Medicine* **22**:2311-2318 https://doi.org/10.1111/jcmm.13519 | PubMed

56 **Rogalska M. E.,** et al. (2024) Transcriptome-wide splicing network reveals specialized regulatory functions of the core spliceosome. *Science* **386**:551-560 https://doi.org/10.1126/science.adn8105 | PubMed

57 **Shyu A. B.,** Wilkinson M. F., Van Hoof A. (2008) Messenger RNA regulation: to translate or to degrade. *The EMBO journal* **27**:471-481 https://doi.org/10.1038/sj.emboj.7601977 | PubMed

58 **Feingold E.,** et al. (2004) The ENCODE (ENCyclopedia of DNA elements) project. *Science* **306**:636-640 https://doi.org/10.1126/science.1105136

59 **Lachmann A.,** et al. (2010) ChEA: transcription factor regulation inferred from integrating genome-wide ChIP-X experiments. *Bioinformatics* **26**:2438-2444 https://doi.org/10.1093/bioinformatics/btq466 | PubMed

60 **Ge M.,** Miao J., Qi J., Zhou X., Lin Z. (2025) TIVelo: RNA velocity estimation leveraging cluster-level trajectory inference. *Nature Communications* **16**:6258 https://doi.org/10.1038/s41467-025-61628-x | PubMed

61 **Qiao C.,** Huang Y. (2021) Representation learning of RNA velocity reveals robust cell transitions. *Proceedings of the National Academy of Sciences* **118**:e2105859118 https://doi.org/10.1073/pnas.2105859118 | PubMed

62 **Bastidas-Ponce A.,** et al. (2019) Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. *Development* **146**:dev173849 https://doi.org/10.1242/dev.173849 | PubMed

Peer reviews

Reviewer #1 (Public review):

Summary:

In the paper, the authors propose a new RNA velocity method, TSvelo, which predicts the transcription rate linearly based on the expression of RNA levels of transcription factors. This framework is an extension of its recent work TFvelo by including unspliced reads and designing a coherent neuralODE framework. Improved performance was demonstrated in six diverse datasets.

Strengths:

Overall, this method introduces innovative solutions to link cell differentiation and gene regulation, with a balance between model complexity (neuralODE) and interpretability (raw gene space).

Comments on revised version:

The authors have added comprehensive analyses in this revision, and all of my concerns have been very well addressed. Here, I just want to re-emphasize the original points 1 and 3.

(1) The analysis and clarification are very helpful - thanks! I found that Fig. R1 and R2 are very insightful, as DoRothEA-only returns much worse performance. Please consider adding these two figures to the supp figure and possibly highlighting your setting for edge pruning (down-weights); therefore, the model is more likely to be affected by false negatives than false positives in the TF-target prior.

(3) Please consider adding some discussion on the challenges in capturing cell cycle transitions.

https://doi.org/10.7554/eLife.108950.2.sa2

Reviewer #3 (Public review):

Despite the abundance of RNA velocity tools, there are still major limitations, and there is strong skepticism about the results these methods lead to. In this paper, the authors try to address some limitations of current RNA velocity approaches by proposing a unified framework to jointly infer transcriptional and splicing dynamics. The method is then benchmarked on 6 real datasets against the most popular RNA velocity tools.

Comments on revised version.

The Authors addressed all my comments suitably. I'd like to thank them for the time they spent addressing them: the revised paper is much more convincing.

I have 2 very minor follow-up concerns:

(1) I appreciated the simulation study, however, no null simulation is present.

We know RNA velocity tools are inclined to provide false positives: trajectories even when the data doesn't have any.

I'd be helpful to add null simulations where the data has no trajectories and see if methods erroneously identify any.

(2) Several of the novel analyses are only reported in the Supplementary material and only references in the main text (e.g., "A validation of TSvelo on simulated data is provided in Fig. S1 and Fig. S2 in the Supplementary Information."). This is pity!

If allowed, I'd add some comments about the new analyses (simulations, computational benchmarks, etc...) also in the main text.

https://doi.org/10.7554/eLife.108950.2.sa1

Author response:

The following is the authors' response to the original reviews.

> ***Public Reviews:***
>
> ***Reviewer #1 (Public review):***
>
> *Summary:*
>
> *In the paper, the authors propose a new RNA velocity method, TSvelo, which predicts the transcription rate linearly based on the expression of RNA levels of transcription factors. This framework is an extension of its recent work TFvelo by including unspliced reads and designing a coherent neuralODE framework. Improved performance was demonstrated in six diverse datasets.*
>
> *Strengths:*
>
> *Overall, this method introduces innovative solutions to link cell differentiation and gene regulation, with a balance between model complexity (neuralODE) and interpretability (raw gene space).*

We thank the reviewer for the positive evaluation of our work and for recognizing the novelty of the proposed framework. We appreciate the reviewer's summary highlighting that TSvelo extends our previous method TFvelo by incorporating unspliced reads and introducing a coherent neuralODE framework to model transcription dynamics.

We are encouraged that the reviewer recognizes the potential of our approach to link cell differentiation with gene regulatory mechanisms, while maintaining a balance between model expressiveness and interpretability in the gene expression space. In the revised manuscript, we have further clarified several methodological details and strengthened the presentation to better highlight these aspects.

> *Weaknesses:*
>
> *While it seems to provide convincing results, there are multiple technical concerns for the authors to clarify and double-check.*
>
> *(1) The authors should clarify and discuss the TF-target map: here, the TF-target genes map is predefined by the TF binding's ChIP-seq data. This annotation is largely incomplete and mostly compiled from a set of bulk tissues. Therefore, for a certain population, the TF-target relation may change. This requires clarification and discussion, possibly exploring how to address this in the model. In addition, a regulon database could be added, e.g., DoRothEA?*

We thank the reviewer for this important comment. The TF–target maps used in TSvelo (e.g., derived from ChIP-seq-based resources such as ENCODE) reflect aggregated TF binding evidence collected across diverse bulk cell types and experimental conditions. As such, they are inherently incomplete and do not capture fully context-specific regulatory activity in a

given primary tissue. In TSvelo, we therefore do not treat these annotations as fixed or cell-type-specific ground truth regulatory relationships. Instead, they are used as a permissive prior that encodes a broad set of potential regulatory interactions.

Within the TSvelo framework, the contribution of each TF–target interaction is learned from data through weight estimation, allowing the model to down-weight or effectively ignore prior edges that are inconsistent with the observed single-cell expression dynamics. This design enables TSvelo to remain robust even when the prior TF–target map is noisy, incomplete, or derived from heterogeneous bulk contexts.

Following the reviewer's suggestion, we additionally incorporated the DoRothEA regulon database as an alternative prior with confidence-level filtering. We further performed ablation studies on the pancreas dataset and the gastrulation erythroid dataset using different TF–target resources, including ChEA, ENCODE, and their combinations with DoRothEA.

The results on the pancreas dataset and the gastrulation erythroid dataset are shown in Figure S13 and Figure S14 respectively, which come up with the same conclusion. We observed highly consistent results across most TF–target prior combinations, including ChEA, ENCODE, ChEA+ENCODE, ChEA+DoRothEA, ENCODE+DoRothEA, and ChEA+ENCODE+DoRothEA. Using the pancreas dataset as example, the mean velocity consistency ranged from 0.985 to 0.995, the mean in-cluster coherence ranged from 0.983 to 0.992, and the mean cross-boundary direction correctness ranged from 0.719 to 0.740 across all settings. These consistently high and tightly bounded metrics indicate that TSvelo is largely insensitive to the specific choice of TF–target prior.

The only configuration showing reduced stability was the use of DoRothEA alone, particularly in terms of cross-boundary direction correctness. This is likely due to its comparatively limited coverage of TF–target interactions. For instance, in the pancreas dataset, only 81 out of 2000 highly variable genes (HVGs) could be associated with TFs based on DoRothEA, corresponding to 102 TF–target links in total, which may restrict downstream regulatory modeling. In contrast, ChEA covered 1793 genes with 13,976 TF–target links, and ENCODE covered 1854 genes with 33,076 links. These results further suggest that integrating multiple TF–target resources could improve performance, likely due to increased coverage and complementary regulatory information.

We further acknowledge that regulatory interactions are inherently context-dependent, and that no static TF–target resource can fully capture tissue-specific regulatory programs. In the revised Discussion, we explicitly clarify this limitation and highlight that incorporating context-specific regulatory data (e.g., single-cell chromatin accessibility or perturbation-based regulatory maps) represents an important direction for future improvement.

> *(2) The authors should clarify how example genes are selected. This is particularly unclear in Figure 2d.*

We thank the reviewer for raising this point. The example genes shown in Fig. 2d were selected to illustrate representative scenarios where our method provides advantages, particularly cases in which the unspliced–spliced 2D phase portrait exhibits mixed or overlapping patterns that are difficult to model using conventional RNA velocity approaches. These examples are therefore intended to demonstrate the types of transcriptional dynamics that TSvelo is designed to better capture.

To avoid the impression of selective presentation, we note that our conclusions are based on systematic evaluation across all genes and datasets. Additional visualizations for a broader set of genes on this dataset are provided in Fig. S3. We have clarified the example gene selection criteria in the revised manuscript.

> *(3) The authors should clarify confidence in the statement in lines 179-180, that ANXA4 should initially decrease. This is particularly concerning, as TSvelo didn't capture the cell cycle transitions well during the initial part.*

We thank the reviewer for raising this point. The statement that ANXA4 initially decreases is based on the observed expression pattern in the dataset rather than on cell-cycle–related dynamics inferred by the model. Specifically, ANXA4 shows higher expression in Ductal cells compared to Ngn3 EP cells, and Ductal represents an earlier stage in the developmental trajectory. Therefore, along the Ductal to Ngn3 EP transition, ANXA4 naturally exhibits an initial decrease in expression. We have clarified this point in the revised manuscript.

> *(4) A support reference should be added for the statement in line 260 that "neuron migrations are inside-out manner". There is no reference supporting this, and this statement is critical for the model assessment.*

We thank the reviewer for this suggestion. This pattern has been reported in previous studies [1,2], which have been added into the revised manuscript.

To Improve clarity, we have also revised the statement in the manuscript as follows:

“During cortical development, neurons follow an inside-out layering pattern in which earlier-born neurons populate the deep cortical layers, whereas later-born neurons migrate past them to occupy more superficial layers.”

(1) Nadarajah, B., Parnavelas, J. Modes of neuronal migration in the developing cerebral cortex. Nat Rev Neurosci 3, 423–432 (2002).

(2) Li, C., Virgilio, M.C., Collins, K.L. et al. Multi-omic single-cell velocity models epigenome–transcriptome interactions and improves cell fate prediction. Nat Biotechnol 41, 387–398 (2023).

> *(5) The comparison to scMultiomics data is particularly interesting, as MultiVelo uses ATAC data to predict the transcription rate. It would be very insightful to add a direct comparison of the estimated transcription rate between using ATAC and directly using TFs' RNA expressions.*

We thank the reviewer for suggesting this highly interesting comparison between ATAC-derived regulatory activity and TF RNA-based proxies for transcription rate estimation.

We have conducted the requested analysis by computing gene-wise chrome accessibility rate used in MultiVelo and the learned transcription rate from TSvelo, and evaluated their correlation across genes. As shown in Figure S15, the two estimates exhibit almost no global correlation across genes, indicating that they capture substantially different aspects of regulatory information.

This discrepancy is not unexpected and reflects the fundamental differences between these modalities. scATAC-seq measures chromatin accessibility, which provides a proxy for cis-regulatory potential of genomic regions. However, ATAC signals are inherently sparse and often exhibit a near-binary structure, limiting their ability to directly capture fine-grained temporal regulatory dynamics. In contrast, TF RNA expression reflects downstream transcriptional output, which is shaped by multiple regulatory layers, including post-transcriptional regulation, protein activity, temporal delays, and indirect regulation through intermediate transcriptional or signaling pathways. As a result, these two modalities are expected to capture complementary but not directly comparable aspects of gene regulation.

Overall, this result suggests that ATAC-based and TF RNA-based signals capture distinct aspects of gene regulation. This further implies that integrating both modalities may be

beneficial for future models that aim to more comprehensively characterize transcriptional regulation. We have added this discussion to the supplementary information.

> *(6) In Figure 6g, it should be clarified how the lineage was determined. Did the authors use the LARRY barcodes, predicted cell fate, or any other methods? Here, the best way is probably using the LARRY barcodes for individual clones.*

We thank the reviewer for this suggestion. The lineage assignment used in Fig. 6g is described in the Methods section ("Lineage segmentation and pseudotime initialization"). Briefly, lineages are inferred from the transcriptomic structure of the data by performing Leiden clustering followed by PAGA-based connectivity analysis. Starting from an initial Leiden cluster, the filtered PAGA graph defines the shortest paths to other clusters, which are considered as the detected lineages, and diffusion pseudotime (DPT) is then used to initialize pseudotime along each lineage. Thus, in this analysis lineages are determined from the expression-derived trajectory structure. We have clarified this point in the revised manuscript and refer readers to the Methods section.

> ***Reviewer #2 (Public review):***
>
> *Summary:*
>
> *Li et al. propose TSvelo, a computational framework for RNA velocity inference that models transcriptional regulation and gene-specific splicing using a neural ODE approach. The method is intended to improve trajectory reconstruction and capture dynamic gene expression changes in scRNA-seq data. However, the manuscript in its current form falls short in several critical areas, including rigorous validation, quantitative benchmarking, clarity of definitions, proper use of prior knowledge, and interpretive caution. Many of the authors' claims are not fully supported by the evidence.*

We thank the reviewer for the careful evaluation of our manuscript and for the constructive comments. We appreciate the concerns regarding validation, benchmarking, methodological clarity, and interpretation. In the revised manuscript, we have carefully addressed these points by adding additional analyses, clarifying methodological details, and moderating several claims to ensure they are fully supported by the data. Detailed responses to each comment are provided below.

> *Major comments:*
>
> *(1) Modeling comments*
>
> *(a) Lines 512-513: How does the U-to-S delay validate the accuracy of pseudotime? Using only a single gene as an example is not sufficient for "validation."*

We thank the reviewer for this important clarification. In the revised manuscript, we have rephrased this part to clarify that Fig. 1a serves only as an illustrative example showing the U-to-S delay for a single gene. Accordingly, we have corrected our statement to indicate that the U-to-S delay is used to infer trajectory orientation, rather than to validate the accuracy of pseudotime.

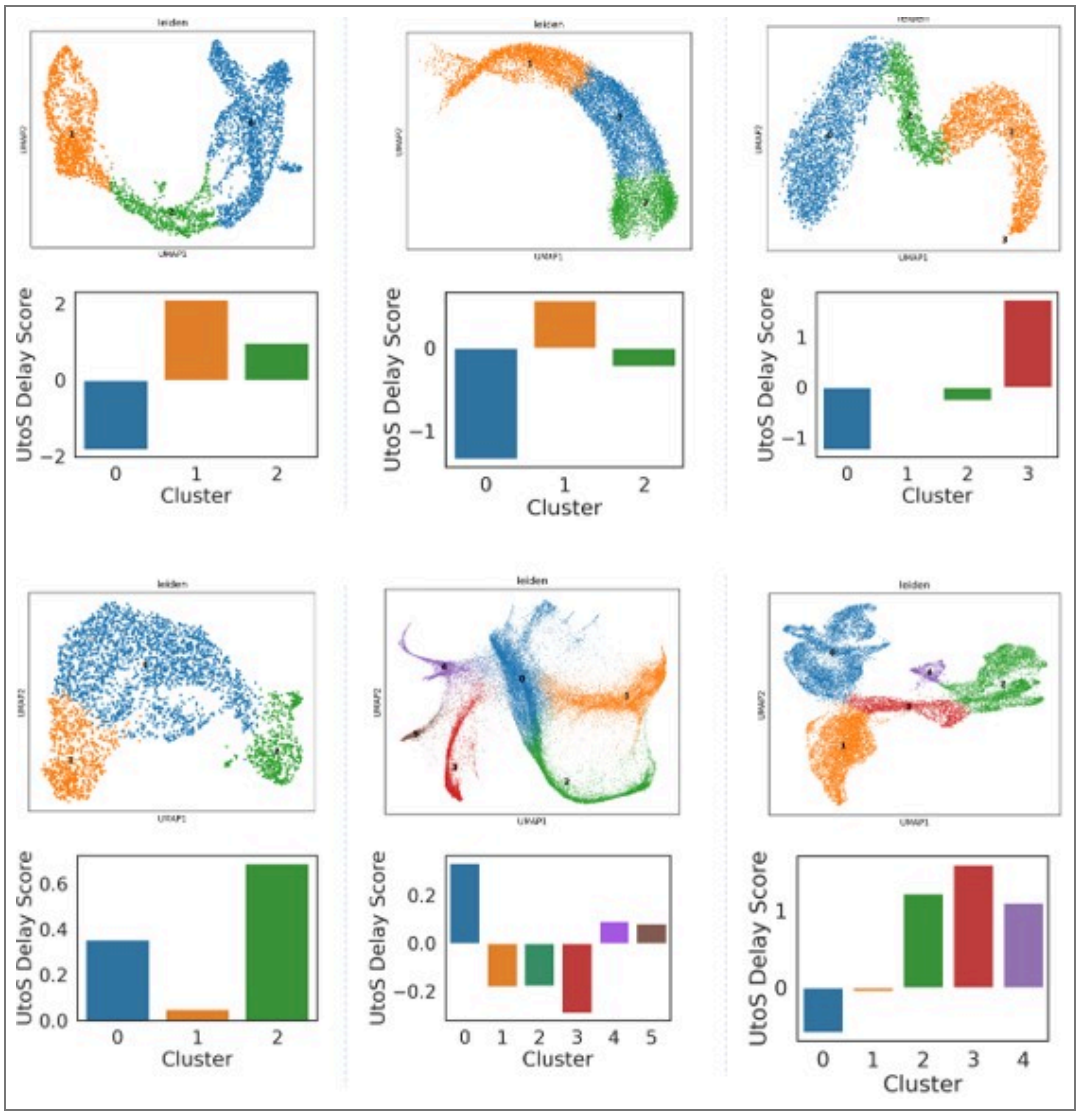
In addition, we have expanded the description to explain that U-to-S delay signals are aggregated across all genes to provide a more robust and comprehensive assessment for this purpose. Additional analysis is provided in our response to the next comment.

> *(b) Lines 512-518: The authors propose a strategy for selecting the initial state, but do not benchmark how accurate this selection procedure is, nor do they provide sufficient rationale. While some genes may indeed exhibit U-to-S delay during lineage differentiation, why does the highest U-to-S delay score indicate the correct initiation*

> *states? Please provide mathematical justification and demonstrate accuracy beyond using a single gene example. Maybe a simulation with ground truth could help here, too.*

We thank the reviewer for this insightful comment. In the revised manuscript, we have clarified both the intuition and justification of this approach. Briefly, along a correctly oriented trajectory, unspliced (U) expression is expected to precede spliced (S) expression due to transcriptional dynamics. Ideally, this U-to-S delay would be observable at the level of individual genes. However, due to the high noise inherent in scRNA-seq data, such delays are often not consistently detectable on a per-gene basis. To address this, we aggregate U-to-S delay signals across all genes and determine the lineage orientation by maximizing a global delay score. Under this criterion, the cluster from which all outgoing lineages exhibit the highest aggregated U-to-S delay is inferred to correspond to the initial state.

We emphasize that this approach relies on genome-wide aggregation rather than any single gene. Moreover, the same strategy is applied uniformly across all six datasets using identical parameter settings, demonstrating its robustness and stability. To further address the reviewer's concern, we additionally present the U-to-S delay scores for each Leiden cluster when treated as the initial state across all datasets (Author response image 1). The results on all datasets suggest that the highest U-to-S delay scores can be used to detect the initial cluster.



Author response image 1. The U-to-S delay scores for each Leiden cluster when treated as the initial state across all datasets.

Following your suggestions, we also add a simulation study. We generated synthetic single-cell RNA velocity datasets using a mechanistic transcriptional dynamics model with one or multiple developmental branches. The system included 200 genes, among which 30 were designated as transcription factors (TFs).

For each branch, we independently sampled a TF–target regulatory matrix $W \in R^{30\times200}$ from a standard normal distribution to simulate distinct GRN structures. Gene expression dynamics were modeled using a coupled ordinary differential equation (ODE) system describing unspliced and spliced RNA abundances:

$$\frac{du_g(t)}{dt} = \alpha_g(t) - \beta_g u_g(t)$$
$$\frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t)$$

where u and s denote unspliced and spliced RNA levels, respectively. The transcription rate α was computed as a nonlinear function of TF expression, defined as a weighted sum of spliced TF abundance, followed by clipping to ensure bounded activation.

Each branch is initialized from the same randomly sampled initial condition drawn from a gamma distribution, allowing controlled divergence of trajectories driven solely by branch-

specific regulatory programs.

To simulate observed sequencing counts, we introduced technical noise by scaling latent expression levels with cell-specific library sizes drawn from a log-normal distribution. The resulting expression counts were generated using a negative binomial sampling model:

$$\times x, \theta)$$

where θ controls over dispersion, with smaller values corresponding to higher noise levels. The final datasets consist of paired unspliced (U) and spliced (S) count matrices with realistic transcriptional stochasticity and branching gene regulatory dynamics. For each branch, cells were further divided into three developmental stages for downstream analysis.

We evaluated TSvelo on multiple simulated datasets with varying numbers of branches and noise levels. There are two or three branches start from the same root cell groups in these datasets (Branch 1: stage 0 - stage 1 - stage 2. Branch 2: stage 0 - stage 3 - stage 4. Branch 3: stage 0 - stage 5 - stage 6). The results of initial state identification based on the unspliced-to-spliced (U-to-S) delay, along with the corresponding 2D velocity stream visualizations, are presented in Supplementary Figure S1. These results demonstrate that the U-to-S delay–based initialization is robust and consistently identifies cells corresponding to the earliest developmental stage ("stage 0") across different simulation settings. All additional results have been included in the Supplementary Information.

> *(c) Equation (8): The formulation looks to be incorrect. If $$W \in \mathbb{R}^{G\times G}$$ and $$W' - \Gamma' \in \mathbb{R}^{K\times K}$$, how can they be aligned within the same row? Please clarify.*

We thank the reviewer for pointing this out. This was a typographical error in the manuscript. In the third line of Equation (8), the term should be *W'* instead of *W*. We have corrected this in the revised manuscript to ensure dimensional consistency.

$$\begin{bmatrix} dU/dt \\ dS/dt \\ dS'/dt \end{bmatrix} = A \begin{bmatrix} U \\ S \\ S' \end{bmatrix} = \begin{bmatrix} -B & W & W' \\ B & -\Gamma & 0 \\ 0 & W' & W' - \Gamma' \end{bmatrix} \begin{bmatrix} U \\ S \\ S' \end{bmatrix}$$

> *(d) The use of prior knowledge graphs from ENCODE or ChEA to constrain regulation raises concerns. Much of the regulatory information in these databases comes from cell lines. How can such cell-line-based regulation be reliably applied to primary tissues, as is done throughout the manuscript? Additional experiments are needed to test the robustness of TSvelo with respect to prior knowledge.*

We thank the reviewer for this important comment. In TSvelo, TF–target networks from resources such as ENCODE and ChEA are incorporated as priors that guide the model toward biologically plausible regulatory structures. Importantly, the contribution of each TF–target interaction is learned from the data, allowing the model to down-weight or override potentially inaccurate or context-mismatched regulatory links. By aggregating signals across a large number of genes, the model further reduces sensitivity to noise and incompleteness in any single prior network.

To evaluate robustness with respect to prior knowledge, we incorporated the DoRothEA regulon resource as an alternative TF–target prior with confidence-level filtering. We further performed ablation studies on the pancreas dataset and the gastrulation erythroid dataset using different TF–target resources, including ChEA, ENCODE, and their combinations with DoRothEA.

The results on the pancreas dataset and the gastrulation erythroid dataset are shown in Figure S13 and Figure S14 respectively, which come up with the same conclusion. We observed highly consistent results across most TF–target prior combinations, including ChEA,

ENCODE, ChEA+ENCODE, ChEA+DoRothEA, ENCODE+DoRothEA, and ChEA+ENCODE+DoRothEA. Using the pancreas dataset as example, the mean velocity consistency ranged from 0.985 to 0.995, the mean in-cluster coherence ranged from 0.983 to 0.992, and the mean cross-boundary direction correctness ranged from 0.719 to 0.740 across all settings. These consistently high and tightly bounded metrics indicate that TSvelo is largely insensitive to the specific choice of TF–target prior. Notably, these results further suggest that even when the underlying regulatory resources differ in origin (e.g., cell-line-derived vs. curated or aggregated datasets), the inferred dynamics remain stable.

The only configuration showing reduced stability was the use of DoRothEA alone, particularly for cross-boundary direction correctness. This is likely due to its comparatively limited coverage of TF–target interactions. For instance, in the pancreas dataset, only 81 out of 2000 highly variable genes (HVGs) could be associated with TFs based on DoRothEA, corresponding to 102 TF–target links in total, which may limit downstream regulatory modeling. In contrast, ChEA covered 1793 genes with 13,976 TF–target links, and ENCODE covered 1854 genes with 33,076 links. These results further suggest that integrating multiple TF–target resources can improve performance, likely due to increased coverage and complementary regulatory information.

We agree that regulatory interactions derived from resources such as ENCODE and ChEA may not fully generalize to primary tissues due to their context-dependent nature. In the revised Discussion, we explicitly clarify this limitation, particularly their inability to capture tissue-specific regulatory programs. We further highlight that incorporating context-specific regulatory data, such as single-cell chromatin accessibility or perturbation-based regulatory maps, represents an important direction for future improvement.

> *(e) Lines 579-580: How is the grid search performed? More methodological details are required. If an existing method was used, please provide a citation.*

The grid search for the time step means that the model evaluates the loss in equation (10) across all candidate values of t_{step} in the set {0,1,2,...,999}. This strategy was originally adopted in scVelo for optimizing the time step parameter. We have now added the corresponding citation to scVelo in the revised manuscript.

> *(2) Application on pancreatic endocrine datasets*
>
> *(a) Lines 140-141: What is the definition of the final pseudotime-fitted time t or velocity pseudotime?*

There is no distinction between “final pseudotime”, “fitted time t” and “velocity pseudotime”. All of them refer to the same quantity in our framework. To eliminate any potential ambiguity, we have standardized the terminology by replacing “final pseudotime” with “pseudotime”.

> *(b) Lines 143-144: The use of the velocity consistency metric to benchmark methods in multi-lineage datasets is incorrect. In multi-lineage differentiation systems, cells (e.g., those in fate priming stages) may inherently show inconsistency in their velocity. Thus, it is difficult to distinguish inconsistency caused by estimation error from that arising from biological signals. Velocity consistency metrics are only appropriate in systems with unidirectional trajectories (e.g., cell cycling). The abnormally high consistency values here raise concerns about whether the estimated velocities meaningfully capture lineage differences.*

We thank the reviewer for raising this important point regarding the use of the velocity consistency metric in multi-lineage systems. Velocity consistency was initially introduced by scVelo [1] and implemented as scvelo.velocity_confidence() in its package. Velocity consistency provides one of the few widely adopted quantitative criteria for benchmarking

RNA velocities [2]. We agree that it is especially suitable for single-lineage processes. For datasets with clear multi-lineage differentiation (Fig. 5 and Fig. 6), we do not use this metric, precisely to avoid the issue highlighted by the reviewer.

However, the pancreatic endocrine dataset (Fig. 2) exhibits minimal branching, making velocity consistency be more appropriate. As introduced by veloVI study, RNA velocities are supposed to change smoothly over the phenotypic manifold [3]. Higher consistency indicates that neighboring cells show compatible velocity directions, reflecting stable and coherence of the inferred velocity field. Additionally, multiple previous studies used velocity consistency to evaluate model performance on this pancreas dataset [2,3,4], providing a standard point of comparison.

To better address your concerns, we have replaced the corresponding panel in Fig. 2 of the main text with an evaluation of cell-type separability in both the traditional 2D (unspliced–spliced) phase portrait and the learned 3D (α–unspliced–spliced) phase portrait by TSvelo (Author response image 4 in our response to your subsequent question). We appreciate your suggestions, as the comparison more clearly highlights the novelty and contribution of TSvelo and helps explain its improved performance. Now, the velocity consistency panel has been moved to the Supplementary Information. In addition, we have added a clearer explanation of the cross-boundary correctness metric in the revised manuscript.

(1) Bergen, V., Lange, M., Peidli, S., Wolf, F. A., & Theis, F. J. (2020). Generalizing RNA velocity to transient cell states through dynamical modeling. Nature Biotechnology, 38(12), 1408-1414.

(2) Luo, Y., Ren, J., Yang, Q. ... & Li, Q. (2026). Benchmarking RNA velocity methods across 17 independent studies, Cell Reports Methods, 101367.

(3) Gayoso, A., Weiler, P., Lotfollahi, M., Klein, D., Hong, J., Streets, A., ... & Yosef, N. (2024). Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. Nature Methods, 21(1), 50-59.

(4) Li, J., Pan, X., Yuan, Y., & Shen, H. B. (2024). TFvelo: gene regulation inspired RNA velocity estimation. Nature Communications, 15(1), 1387.

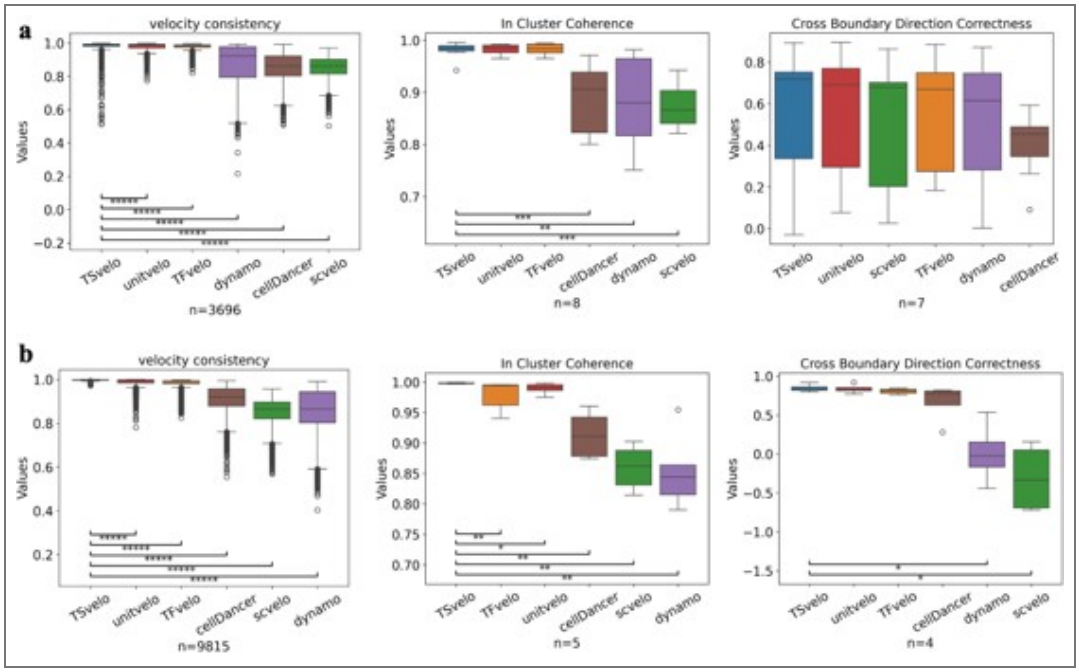
> *(c) The improvement of TSvelo over other methods in terms of cross-boundary direction correctness looks marginal; a statistical test would help to assess its significance.*

We thank the reviewer for this insightful comment. In the revised manuscript, we have added statistical tests for evaluated metrics, including velocity consistency, cross-boundary direction correctness, and in-cluster coherence.

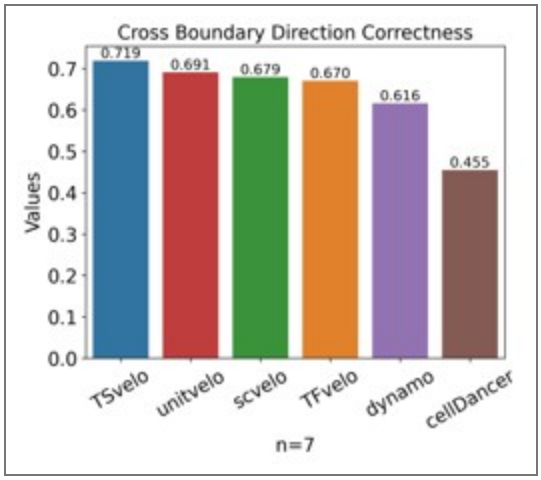
As shown in Author response image 2, TSvelo significantly outperforms all baseline methods in terms of velocity consistency across both datasets. For in-cluster coherence, TSvelo achieves significantly better performance on the gastrulation (erythroid) dataset, while on the pancreas dataset it performs comparably to the best-performing baselines (UniTVelo and TFvelo) and significantly outperforms several competing methods, including CellDancer, Dynamo, and scVelo.

For cross-boundary direction correctness, TSvelo shows consistent improvements in mean performance on the pancreas dataset (Author response image 3), and significantly outperforms Dynamo and scVelo on the gastrulation dataset. Although not all pairwise comparisons on cross-boundary direction correctness reach statistical significance, this is likely influenced by the limited number of independent samples (n = 7 and n = 4 for the two datasets, respectively), which reduces statistical power for detecting differences. Importantly, TSvelo still achieves the best average performance among all methods, indicating a consistent overall trend in favor of TSvelo.

We have added these results into the revised manuscript.



Author response image 2. The quantitative comparison between TSvelo and baseline approaches on the pancreas dataset (panel a) and the gastrulation erythroid dataset (panel b). In each plot, methods are ranked in descending order of their mean values. Numbers at the bottom indicate the sample size for each metric. Significance is determined using a one-sided Mann–Whitney U test. *****, ***, ** and * represent $p < 0.00001$, $0.0001 \le p < 0.001$, $0.001 \le p < 0.01$, and $0.01 \le p < 0.05$, respectively.



Author response image 3. The comparison of mean cross-boundary direction correctness on the pancreas dataset.

> *(d) Lines 177-178: Based on the figure, TSvelo does not appear to clearly distinguish cell types. A quantitative metric, such as Adjusted Rand Index (ARI), should be provided.*

We thank the reviewer for this helpful suggestion. To quantitatively assess whether TSvelo can distinguish cell types, we evaluated the separability of cell-type labels in both the 2D

(unspliced–spliced) phase portrait adopted by previous RNA velocity approaches, and the 3D (α–unspliced–spliced, α denotes the transcriptional rate) phase portrait introduced by TSvelo.

Specifically, we evaluated how well the embedding preserves cell-type information using a k-nearest neighbors (kNN) classification accuracy with 5-fold cross-validation. Given an embedding matrix in 2D or 3D space ($X \in \mathbb{R}^{n*d}$, where n is the number of cells and d is 2 or 3) and corresponding cell-type labels ($y \in \{1, \ldots, C\}$, we partition the data into five folds. For each fold (k), a kNN classifier with $K = 5$, denoted as $f^{(k)}$, is trained on the training subset and evaluated on the held-out test subset. The classification accuracy for the k-th fold is defined as $\mathbb{R}$

$$Acc^{(k)} = \frac{1}{n_k} \sum_{i \in D_k^{\text{test}}} \mathbf{1}\left(f^{(k)}(x_i) = y_i\right)$$

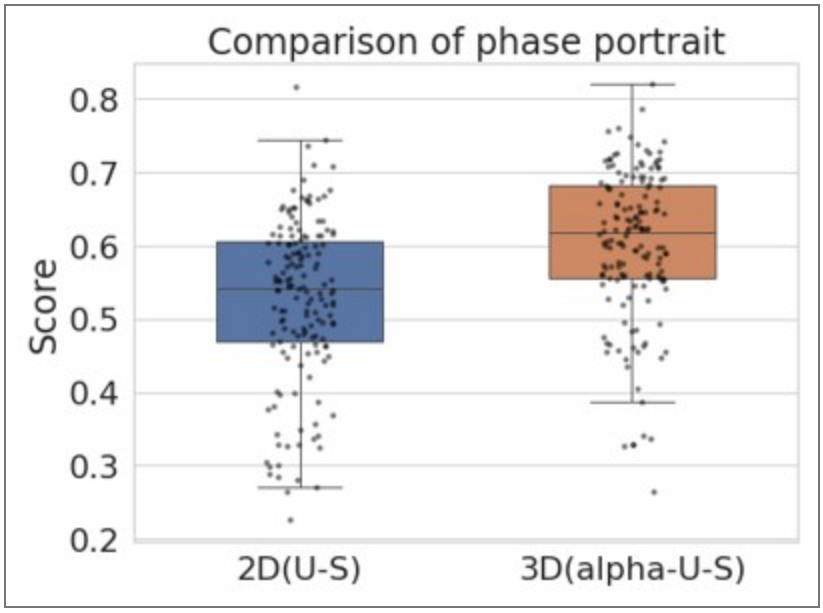
where n_k is the number of samples in the test set and 1(.)is the indicator function. The final score is obtained by averaging across all folds:

$$Acc = \frac{1}{5} \sum_{k=1}^{5} Acc^{(k)}$$

This metric directly assesses whether cells of the same type are positioned close to each other in the embedding space, and is widely used to quantify representation quality.

Using this evaluation, we observed that the 3D phase portrait consistently achieves significantly higher accuracy than the 2D phase portrait (Author response image 4). The improvement is highly statistically significant (one-sided Mann–Whitney U test, p-value = 4.37×10^{-10}), demonstrating that the 3D representation provides substantially better separation of cell types.

We have added these quantitative results to the revised manuscript to complement the visual evidence and to clarify that TSvelo effectively distinguishes cell types in the learned representation.



Author response image 4. **The evaluation of the separability of cell-type labels in both the 2D (unspliced–spliced) phase portrait and the 3D (α–unspliced–spliced) phase portrait for the pancreas dataset.**

> *(e) Lines 179-183: The claim that traditional methods cannot capture dynamics in the unspliced-spliced phase portrait is vague. What specific aspect is not captured-the fitted values or something else? Evidence is lacking. Please provide a detailed explanation and quantitative metrics to support this claim.*

We thank the reviewer for this important comment. We have revised the text to more clearly illustrate this point using representative example genes as follows: “For instance, ANXA4 shows higher expression in Ductal cells compared to Ngn3 low EP cells, which mean its expression pattern exhibits an initial decrease followed by an increase. Such dynamics are not easily captured in the conventional unspliced–spliced phase portrait used by previous approaches, as many baseline methods implicitly assume a decreasing–then–increasing expression pattern. By comparison, TSvelo can still fit such expression pattern by using additional information from the 3D phase portrait.”

In addition, we also clarify that the 2D u–s representation has limited capacity to separate heterogeneous dynamic cell states, which can affect downstream velocity field estimation. In the conventional 2D u–s phase portrait, cells from different dynamic regimes may overlap in the same region of the embedding space. This overlap reduces the identifiability of underlying transcriptional states and makes the inferred local dynamics more ambiguous. In contrast, TSvelo introduces an additional latent variable α, forming a 3D (α, u, s) phase portrait, which helps disentangle these mixed trajectories and yields a more structured and separable representation of cell dynamics. We have provided quantitative evidence in the previous response (Author response image 4). Briefly, the proposed 3D representation achieves consistently higher kNN classification accuracy (5-fold cross-validation, k=5) for cell state identification compared to the 2D u–s embedding.

> *(3) Application to gastrulation erythroid datasets*
>
> *(a) Lines 191-194: The observation that velocity genes are enriched for erythropoiesis-related pathways is trivial, since the analysis is restricted to highly variable genes (HVGs) from an erythropoiesis dataset. This enrichment is expected and therefore not informative.*

We thank the reviewer for this comment and agree that such enrichment is expected given the use of HVGs from an erythropoiesis dataset. This analysis was included only as a preliminary sanity check to support the plausibility of the inferred velocity genes, rather than as a main result. We have accordingly simplified the description and clarified that this analysis serves only as a preliminary check in the revised manuscript.

> *(b) Lines 227-228: It remains unclear how TSvelo "accurately captures the dynamics." What is the definition of dynamics in this context? Figure 3g shows unspliced/spliced vs. fitted time plots and phase portraits, but without a quantitative definition or measure, the claim of superiority cannot be supported. Visualization of a single gene is insufficient; a systematic and quantitative analysis is needed.*

We thank the reviewer for this important comment. We have revised the text to more clearly illustrate this point using representative example genes as follows: “For HSP90AB1, which exhibits a counter-clockwise pattern in the unspliced–spliced phase portrait, in contrast to the clockwise dynamics typically assumed by most baseline approaches, it is difficult for previous methods to capture this behavior, whereas TSvelo can still faithfully model such patterns. For genes such as RPS26, which have critical roles in the development in blood progenitors to erythroid40, the unspliced-spliced data is so noisy that cells of different types overlap in phase portrait. TSvelo can still captures the gene dynamics and reveals differences in transcription rates across cell types.”

In addition, we explicitly emphasize the role of the 3D (α, u, s) phase portrait, which provides a more structured and separable representation of transcriptional states compared to the conventional 2D u–s space. This improved representation is the key factor underlying the advantages of TSvelo in modeling transcriptional processes. In the conventional 2D u–s phase portrait, cells from different transcriptional states may overlap, leading to reduced separability. In contrast, introducing the latent variable α expands the representation to a 3D space, which helps disentangle these mixed states and yields a clearer phase structure. Similar to our previous response in Author response image 4, we provide quantitative evidence on this gastrulation erythroid dataset in Figure S7, showing that the 3D representation achieves consistently higher kNN classification accuracy for cell state separation compared to the 2D u–s embedding (one-sided Mann–Whitney U test, p-value = 0.002).

> *(4) Application to the mouse brain and other datasets*
>
> *(a) Lines 280-281: The authors cannot claim that velocity streams are smoother in TSvelo than in Multivelo based solely on 2D visualization. Similarly, claiming that one model predicts the correct differentiation trajectory from a 2D projection is over-interpretation, as has been discussed in prior literature see PMID: 37885016.*

We thank the reviewer for this important comment. Consistent with other RNA velocity studies, TSvelo employs the 2D UMAP stream plot for visualizing the results. We agree that conclusions based solely on 2D visualizations may lead to over-interpretation. Our intention was to provide an intuitive visualization rather than a rigorous quantitative comparison. Accordingly, we have revised the text to avoid making definitive claims about smoothness or correctness of differentiation trajectories based solely on 2D projections.

> *(b) Lines 304-306: Beyond transcriptional signal estimation, how is regulation inferred solely from scRNA-seq data validated, especially compared with scATAC-seq data? Are there cases where transcriptome-based regulatory inference is supported by epigenomic evidence, thereby demonstrating TSvelo's GRN inference accuracy?*

We thank the reviewer for this important question regarding the validation of regulatory inference derived from scRNA-seq data and its comparison to scATAC-seq-based evidence.

We would like to first clarify the scope of TSvelo. Similar to existing RNA velocity methods, the primary goal of TSvelo is to model transcriptional dynamics and accurately infer cell state transitions and cell fate trajectories. In this context, gene regulatory information is not inferred de novo from data, but incorporated as prior knowledge from curated TF–target databases to guide and constrain the dynamics modeling process, as described in our Introduction.

We have conducted the requested analysis by computing gene-wise chrome accessibility rate used in MultiVelo and the learned transcription rate from TSvelo, and evaluated their correlation across genes. As shown in Figure S15, the two estimates exhibit almost no global correlation across genes, indicating that they capture substantially different aspects of regulatory information.

This discrepancy is not unexpected and reflects the fundamental differences between these modalities. scATAC-seq measures chromatin accessibility, which provides a proxy for cis-regulatory potential of genomic regions. In contrast, TF RNA expression reflects downstream transcriptional output, which is shaped by multiple regulatory layers, including post-transcriptional regulation, protein activity, temporal delays, and indirect regulation through intermediate transcriptional or signaling pathways. As a result, these two modalities are expected to capture complementary but not directly comparable aspects of gene regulation.

We acknowledge that scATAC-seq provides valuable complementary information on chromatin accessibility and regulatory potential, and will consider incorporating matched multi-omics data in future work. In the revised manuscript, we further clarify that TSvelo is an RNA velocity method that incorporates prior knowledge from curated TF–target databases, and we have added a discussion on the potential use of scATAC-seq data for future extension of our framework.

> *(c) The claim that TSvelo can model multi-lineage datasets hinges on its use of PAGA for lineage segmentation, followed by independent modeling of dynamics within each subset. However, the procedure for merging results across subsets remains unclear.*

We thank the reviewer for pointing out that the merging step was not sufficiently described. After modeling dynamics independently within each lineage-specific subset, TSvelo integrates the results via a weighted aggregation procedure at the cell level.

For each cell and each inferred quantity (e.g., velocity or other dynamic variables), we collect the estimates obtained from different lineage-specific models and combine them using a weighted average. The weights are defined by the size of each lineage, reflecting its statistical support. We have clarified details about this merging procedure in the Methods section.

This aggregation reconciles multiple lineage-specific estimates for the same cell into a single value and mitigates discontinuities that could arise from directly combining independent lineage analyses. The resulting values define a unified set of dynamics for each cell across lineages.

> ***Reviewer #3 (Public review):***
>
> *Despite the abundance of RNA velocity tools, there are still major limitations, and there is strong skepticism about the results these methods lead to. In this paper, the authors try to address some limitations of current RNA velocity approaches by proposing a unified framework to jointly infer transcriptional and splicing dynamics. The method is then benchmarked on 6 real datasets against the most popular RNA velocity tools.*
>
> *While the approach has the potential to be of interest for the field, and may present improvements compared to existing approaches, there are some major limitations that should be addressed, particularly concerning the benchmark (see major comment 1).*
>
> *Major comments:*
>
> *(1) My main criticism concerns the benchmarking: real data lack a ground truth, and are absolutely not ideal for comparing methods, because one can only speculate what results appear to be more plausible.*
>
> *A solid and extensive simulation study, which covers various scenarios and possibly distinct data-generating models, is needed for comparing approaches. The authors should check, for example, the simulation studies in the BayVel approach (Section 4, BayVel: A Bayesian Framework for RNA Velocity Estimation in Single-Cell Transcriptomics). Clearly, all methods should be included in the simulation.*

Following your recommendation, we have added the simulation analysis to compare TSvelo with existing RNA velocity approaches. We generated synthetic single-cell RNA velocity datasets using a mechanistic transcriptional dynamics model with one or multiple developmental branches. The system included 200 genes, among which 30 were designated as transcription factors (TFs).

For each branch, we independently sampled a TF–target regulatory matrix $W \in \mathbb{R}^{30\times200}$ from a standard normal distribution to simulate distinct GRN structures. Gene expression

dynamics were modeled using a coupled ordinary differential equation (ODE) system describing unspliced and spliced RNA abundances:

$$\frac{du_g(t)}{dt} = \alpha_g(t) - \beta_g u_g(t)$$
$$\frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t)$$

where u and s denote unspliced and spliced RNA levels, respectively. The transcription rate α was computed as a nonlinear function of TF expression, defined as a weighted sum of spliced TF abundance, followed by clipping to ensure bounded activation.

Each branch is initialized from the same randomly sampled initial condition drawn from a gamma distribution, allowing controlled divergence of trajectories driven solely by branch-specific regulatory programs.

To simulate observed sequencing counts, we introduced technical noise by scaling latent expression levels with cell-specific library sizes drawn from a log-normal distribution. The resulting expression counts were generated using a negative binomial sampling model:

$$\times x, \theta)$$

where θ controls over dispersion, with smaller values corresponding to higher noise levels. The final datasets consist of paired unspliced (U) and spliced (S) count matrices with realistic transcriptional stochasticity and branching gene regulatory dynamics. For each branch, cells were further divided into three developmental stages for downstream analysis.

We evaluated TSvelo and those splicing-based RNA velocity approaches on multiple simulated datasets with varying numbers of branches and noise levels. There are one, two or three branches start from the same cell group in these datasets (Branch 1: stage 0 - stage 1 - stage 2. Branch 2: stage 0 - stage 3 - stage 4. Branch 3: stage 0 - stage 5 - stage 6). We primarily assessed performance using the cross-boundary direction correctness (CBDir) metric, as it directly evaluates inferred trajectories against ground-truth cell stage annotations, which have been widely adopted in RNA velocity studies such as VeloAE and UniTvelo. In detail, Cross-boundary direction correctness assesses the accuracy of transitions from a source cluster to a target cluster by examining the boundary cells, and requires ground truth annotations. We directly run the function unitvelo.evaluate() provided in UniTVelo to obtain the Cross-boundary direction correctness. In detail, the CBDir is calculated as follows:

$$\mathrm{CBDir}(c) = \frac{1}{|\{c' \in C_A \cap N(c)\}|} \sum_{c' \in C_A \cap N(c)} \frac{\boldsymbol{v}_c \cdot (\boldsymbol{x}_{c'} - \boldsymbol{x}_c)}{|\boldsymbol{v}_c| \cdot |\boldsymbol{x}_{c'} - \boldsymbol{x}_c|}$$

where θ controls over dispersion, with smaller values corresponding to higher noise levels. The final datasets consist of paired unspliced (U) and spliced (S) count matrices with realistic transcriptional stochasticity and branching gene regulatory dynamics. For each branch, cells were further divided into three developmental stages for downstream analysis.

where C_A denotes the set of cells in the target cluster A, and N(c) represents the neighboring cells of a given cell c v_c and x_c denote the low-dimensional velocity and state vectors of cell c, respectively, and $x_{c'}$ denotes the state vector of its neighboring cell.

As shown in Figure S2, TSvelo consistently achieves the highest accuracy across all simulation settings, particularly in scenarios with complex branching structures, which pose significant challenges for baseline methods.

> *(2) Related to the above: since a ground truth is missing, the real data analyses need to be interpreted with caution. I recommend avoiding strong statements, such as "successfully captures the correct gene dynamics", or "accurately infer", in favour of milder statements supported by the data, such as "... aligns with the biological processes described" (as in page 12), or "results are compatible with current biological knowledge", etc...*

We thank the reviewer for this helpful comment. We agree that analyses on real datasets should be interpreted with appropriate caution because definitive ground truth is typically unavailable. Following the reviewer's suggestion, we have revised the wording throughout the manuscript to avoid overly strong claims. For example, statements such as "successfully captures the correct gene dynamics" and "accurately infer" have been replaced with more cautious descriptions such as "consistent with known biological processes".

> *(3) Many methods perform RNA velocity analyses. While there is a brief description, I think it'd be useful to have a schematic summary (e.g., via a Table) of the main conceptual, mathematical, and computational characteristics of each approach.*

We thank the reviewer for this insightful suggestion. We agree that a structured summary of existing RNA velocity methods would improve clarity and accessibility. We have added a new summary table (Table S1) that systematically compares representative RNA velocity approaches in the supplementary information.

> *(4) Related to the above: I struggled to identify the main conceptual novelty of TSvelo, compared to existing approaches. I recommend explaining this aspect more extensively.*

We thank the reviewer for this insightful comment. We agree that the conceptual novelty of TSvelo can be more clearly articulated.

In the revised manuscript, we have expanded the discussion at the beginning of the Results section to explicitly highlight the key distinctions between TSvelo and existing approaches. Specifically, we now clarify that most existing RNA velocity methods predominantly focus on splicing dynamics and typically operate in a gene-wise manner, without capturing coordinated dynamics across genes. In contrast, TSvelo models the full cascade of transcriptional regulation, transcription, and splicing within a unified framework, and estimates RNA velocity jointly across all genes, thereby capturing their coordinated dynamics at the system level.

> *(5) A computational benchmark is missing; I'd appreciate seeing the runtime and memory cost of all methods in a couple of datasets.*

We thank the reviewer for this helpful suggestion regarding computational benchmarking. In the revised manuscript, we have added a systematic comparison of runtime and GPU memory usage across TSvelo and ba methods using simulated datasets of increasing scale (600, 1200, and 1800 cells) on our NVIDIA GeForce RTX 3090 device with 24 GB memory.

Table S2 shows differences in computational efficiency and resource requirements among methods. Specifically, classical methods such as scVelo and Dynamo exhibit very fast runtimes (10–24 seconds) and do not rely on GPU acceleration, reflecting their relatively lightweight modeling strategies. In contrast, deep learning–based approaches, including UniTVelo, cellDancer, and TSvelo, have higher computational costs due to their increased model complexity.

TSvelo exhibits a stable GPU memory footprint (~1.26 GB) across different dataset sizes, indicating that its memory usage is primarily determined by model architecture rather than the number of cells. This level of memory consumption is well within the capacity of modern

GPUs and does not pose practical limitations. In terms of runtime, TSvelo scales approximately linearly with dataset size. The higher computational cost of TSvelo is mainly due to its EM-style optimization procedure, where each M-step also involves multiple optimization updates to infer gene regulatory effects in a global model. This design enables TSvelo to explicitly incorporate regulatory priors and jointly model gene interactions, which is not supported by these baseline methods.

To further improve runtime efficiency, TSvelo allows flexible control of the number of EM iterations. As shown in Figure S16 and Table S3, we evaluated performance under different iteration settings on the simulation dataset. The early stopping strategy employed in the EM framework of TSvelo, which will stop modeling if the loss is not further reduced in the last 3 iterations. Results show that convergence is typically achieved within 3 iterations for this dataset, and increasing the maximum number of iterations beyond this does not further change the results. Notably, even a single iteration already yields competitive performance, likely benefiting from the strong initialization based on unspliced-to-spliced temporal delay.

Overall, these results highlight a trade-off between computational efficiency and modeling expressiveness. While TSvelo is more computationally demanding than classical approaches, it provides a more flexible framework for incorporating regulatory information and capturing complex gene interactions, which we believe justifies the additional computational cost in scenarios requiring accurate dynamical inference.

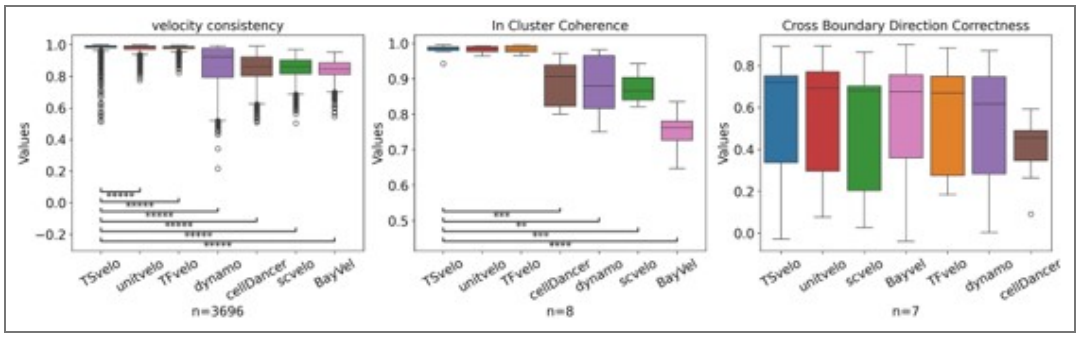
> *(6) I think BayVel (mentioned above) should be added to the list of competing methods (both in the text and in the benchmarks). The package can be found here: https://github.com/elenasabbioni/BayVel_pkgJulia.*

We thank the reviewer for suggesting BayVel and for providing the repository link. We carefully review the available resources, including both the BayVel_pkgJulia and the BayVel_notebooks, and we appreciate the authors' efforts in making their code and data publicly available.

We note that BayVel repositories primarily provide scripts and data for reproducing the figures and results reported in their manuscript. However, at present, the available resources do not yet provide a complete guideline or standardized pipeline for applying BayVel to new datasets. To ensure a fair and reproducible comparison, we therefore tend to use BayVel results officially provided by the authors. We are grateful that the BayVel results on the pancreas dataset is released at BayVel_notebooks page: *https://github.com/elenasabbioni/BayVel_notebooks/tree/main/real%20data/Pancreas/moments/output*.

Based on these results, we conducted comparisons across all methods on the pancreas dataset, with quantitative evaluations shown in Author response image 55. In each plot, methods are ranked in descending order of their mean values. Numbers at the bottom indicate the sample size for each metric. Statistical significance is assessed using a one-sided Mann–Whitney U test, where *****, ***, **, and * denote $p < 0.00001$, $0.0001 \le p < 0.001$, $0.001 \le p < 0.01$, and $0.01 \le p < 0.05$, respectively.

BayVel has now been included in the Introduction, and corresponding comparisons have been added in the revised manuscript.



Author response image 5. The quantitative comparison between TSvelo and baseline approaches on the pancreas dataset. In each plot, methods are ranked in descending order of their mean values. Numbers at the bottom indicate the sample size for each metric. Significance is determined using a one-sided Mann–Whitney U test. *****, ****,***, ** and * represent $p < 0.00001$, $0.00001 \le p < 0.0001$, $0.0001 \le p < 0.001$, $0.001 \le p < 0.01$, and $0.01 \le p < 0.05$, respectively.

> ***Recommendations for the authors:***
>
> ***Reviewer #1 (Recommendations for the authors):***
>
> *Please carefully proofread the text. Some typos:*
>
> *(1) Line 110: differentia -> differential.*
>
> *(2) Line 280: "," to be corrected.*
>
> *(3) Line 566: optimize -> optimizes.*

We thank the reviewer for carefully proofreading the manuscript and for pointing out these typographical errors. We have corrected the identified typos in the revised manuscript.

> ***Reviewer #3 (Recommendations for the authors):***
>
> *(1) Regarding Major Comment 1 in the Public Review, I contacted BayVel authors, who told me that they'll upload all their scripts here within a few days: https://github.com/elenasabbioni/BayVel_notebooks*

Thank you very much for reaching out to the BayVel authors. We sincerely appreciate the BayVel authors' efforts to make their scripts and results publicly available through BayVel_notebooks. We believe this is a valuable contribution that will greatly benefit the community.

We have followed the repository and have now included BayVel in the revised manuscript, with corresponding comparisons added to both the main text and the benchmarking results.

> *(2) Page 9 mentions "consistency", "coherence", and "correctness". Instead of these qualitative (and potentially subjective) evaluations, I'd appreciate using quantitative metrics or visual descriptions when differences are visually clear.*

We thank the reviewer for this insightful comment. The terms "velocity consistency," "in-cluster coherence," and "cross-boundary correctness" used in our manuscript are not intended as subjective descriptions. They correspond to commonly used evaluation criteria in this field and have been adopted as quantitative metrics in previous studies, such as VeloAE[1] and UniTVelo[2]. We have incorporated the following updated definition into the Methods section.

(1) Velocity consistency (VCon). We used the scvelo.velocity_confidence() function from scVelo to evaluate velocity consistency, interpreting the results as a measure of how consistent

velocities are within neighboring cells. Velocity consistency is especially suitable for evaluating the RNA velocity modeling on single lineage. For each cell , the velocity consistency is calculated as follows:

$$\mathrm{VCon}(c) = \frac{1}{|\{c' \in N(c)\}|} \sum_{c' \in N(c)} \frac{\boldsymbol{v}_c \cdot \boldsymbol{v}_{c'}}{|\boldsymbol{v}_c| \cdot |\boldsymbol{v}_{c'}|}$$

Where *N* (c) represents the neighboring cells of a given cell *c* v_c $v_{c'}$ denote the low-dimensional velocity vectors of cell *c*and its neighboring cell *c*'.

(2) Cross-boundary direction correctness (CBDir). Cross-boundary direction correctness assesses the accuracy of transitions from a source cluster to a target cluster by examining the boundary cells, and requires ground truth annotations. We directly run the function unitvelo.evaluate() provided in UniTVelo to obtain the Cross-boundary direction correctness. In detail, the CBDir is calculated as follows:

$$\mathrm{CBDir}(c) = \frac{1}{|\{c' \in C_A \cap N(c)\}|} \sum_{c' \in C_A \cap N(c)} \frac{\boldsymbol{v}_c \cdot (\boldsymbol{x}_{c'} - \boldsymbol{x}_c)}{|\boldsymbol{v}_c| \cdot |\boldsymbol{x}_{c'} - \boldsymbol{x}_c|}$$

Where C_A denotes the set of cells in the target cluster A, and represents the neighboring cells of a given cell *c* v_c $v_{c'}$ denote the low-dimensional velocity and state vectors of cell *c*and its neighboring cell *c*'.

(3) Within-cluster velocity coherence (ICCoh). Within-cluster velocity coherence measures the coherence of velocities within a single cluster using a cosine similarity score between cell velocities. We applied the function unitvelo.evaluate() provided by UniTVelo to directly compute the within-cluster velocity coherence. Using the same notation as defined above, the CBDir is calculated as follows:

$$\mathrm{ICCoh}(c) = \frac{1}{|\{c' \in C_A \cap N(c)\}|} \sum_{c' \in C_A \cap N(c)} \frac{\boldsymbol{v}_c \cdot \boldsymbol{v}_{c'}}{|\boldsymbol{v}_c| \cdot |\boldsymbol{v}_{c'}|}$$

(1) Qiao, C. & Huang, Y. Representation learning of RNA velocity reveals robust cell transitions. Proceedings of the National Academy of Sciences 118, e2105859118 (2021).

(2) Gao, M., Qiao, C. & Huang, Y. UniTVelo: temporally unified RNA velocity reinforces single-cell trajectory inference. Nature Communications 13, 6586 (2022).

> *(3) At page 3, some objects are not defined after formula (3):*
>
> *ReLU finction, and w_gi*
>
> *Additionally, parenthesis of ReLU function should be bigger.*

We thank the reviewer for pointing this out. In the revised manuscript, we have explicitly defined the ReLU activation function and clarified that w_{gi} represents the regulatory weight of TF *i* on the target gene *g*. In addition, we have adjusted the formatting of Eq. (3) by enlarging the parentheses in the ReLU function to improve readability.

https://doi.org/10.7554/eLife.108950.2.sa0