Review History

**First round of review**

**Reviewer 1**

**Were you able to assess all statistics in the manuscript, including the appropriateness of statistical tests used?**

Yes: The presented statistical analyses appear to be sound, and I do not believe additional statistical comparisons are required. Separately, In my full-length review below, I do provide suggested analyses and simulations that I believe will strengthen the manuscript.

**Were you able to directly test the methods?**

Yes

**Comments to author:**

In the manuscript entitled "Mapping lineage-traced cells across time points with Moslin" by Lange\*, Piran\*, Klein\*, Spanjaard\* et al, the authors present a new method for analyzing single-cell lineage tracing (scLT) experiments across multiple time points. This method, termed "moslin", implements an optimal transport (OT) algorithm that regularizes the inference of a coupling matrix across timepoints using lineage information learned from scLT data and paired gene expression data. The authors argue for moslin's effectiveness with both simulated and experimental data, and they propose new putative drivers of C. elegans embryogenesis and biological mechanisms contributing to heart regeneration in a zebrafish model of cardiac injury. In closing, the authors provide a discussion of future extensions to the moslin algorithm, especially how additional modalities may be included.  
Overall, the manuscript describes an elegant, timely, and useful approach for analyzing scLT data. The authors' presentation of the algorithm makes a strong case for its improvement over existing approaches, and also demonstrates how the tool interoperates with the ecosystem of single-cell analysis tools in the scverse. I also commend the authors for providing a thorough and well documented python package, complete with an easy-to-follow tutorial. Together, this work will be of broad interest to the single-cell community, especially those working on the analysis of scLT datasets.   
While I am generally supportive of its publication, I believe the manuscript should be improved with a few additional simulations, analyses, and clarifications before acceptance.  
Major comments:  
1. Depth of simulation experiments. While the authors make a strong case for moslin's effectiveness over other algorithms via simulations, additional simulation experiments should be used to better characterize moslin's performance. The following experiments should be presented, if possible:   
a. A critical step in scLT experiments is inferring phylogenies from barcoding data; while moslin is not a tree inference algorithm, a key question that should be addressed is how errors in tree reconstruction affect moslin's inference. The authors should investigate the relationship between tree reconstruction error (using widely-used statistics like Robinsons-Foulds distance or triplets correct1) and coupling error. The authors can consider scoring the accuracy of tree reconstructions from results presented in Figure 2f. Alternatively, if necessary, to obtain enough variability in tree reconstruction complexity, the authors can reconstruct trees simulated from a spectrum of scLT parameters (e.g., using NJ on TedSim simulated trees).  
b. A hallmark of developmental systems - as observed in later experimental data analyzed - is the presence of transient states only observed at a given timepoint. It is unclear if the authors adequately capture this scenario in their TedSim simulations presented in Fig 1c-f. The authors should either clarify that their simulations generalize to this scenario, or they should study this phenomenon from two perspectives: first, when certain cell states are not observed in the final sampling; second, when there is a large enough temporal gap between samplings such that multiple cell-type transitions might occur, some of which might be transitional cell states not observed at the final timepoint.  
c. While in the simplified simulation framework convergent evolution is explored, it does not appear to be explored in the more sophisticated scLT-based simulations (Figure 2c-f). The authors should provide a TedSim-based simulation experiment in which an amount of convergent evolution is modeled.  
d. Finally, a central challenge to the analysis of scLT data across applications is the bias that may be introduced during sampling. The authors should present a robustness analysis in which error is evaluated against various sampling errors.  
  
2. Analysis of C. elegans embryogenesis. While the authors present a thorough analysis of how moslin improves the analysis of C. elegans embryogenesis, the following should be addressed:  
a. In the section of text on p10 lines 29-34, the authors should more clarify the difference between C. elegans cells with precise lineage information and those without; above, the authors state they decided to focus on the ABpxp lineage because it is well-annotated. What distinguishes the cells with precise lineage and imprecise lineage in ABpxp, also presented in Supplementary Figure 3c?   
b. Legend to Figure 3e: the authors should more clearly state what the "fate probabilities" correspond to, and how they are computed.  
c. It should be stated more clearly why 7 terminal states were chosen in the analysis of C. elegans embryogenesis - is there any dependence on the number of terminal states and the robustness of the presented results?  
  
3. Clarity of Figure 4.  
a. The schematic presented in Figure 4a should be improved with the following: (a) increased resolution of projection and lineage plots on right-hand side; (b) labeling of what the projection axes are; (c) improved annotation of lineage trees.  
b. The authors should present additional display items of the dataset analyzed here, for example the distribution of cell types across time points and individuals.  
c. The heatmaps in Figure 4b should be improved by (a) increasing the resolution of the figure; (b) annotating the colorbar with an informative legend (similarly for Supplementary Figure 17); (c) annotating the dashed lines with an informative label.  
d. The bubbles in Figure 4g should be annotated with Fibroblast cell type names for clarity. Resolution should also be improved.  
  
4. The findings presented in Figure 4 describing a transient col11a1a fibroblast population giving rise to the essential col12a1a fibroblast population is an interesting prediction. Can the authors use the dynamics learned from moslin to speculate on the essentiality of the transient col11a1 population that emerges at 3dpi? Perhaps a synthetic ablation of this population from the dataset might provide additional support that the col11a1a population is an important (or, essential) reservoir giving rise to the col12a1a population?  
  
Minor comments:  
1. In the text describing Figure 2b (p.7 lines 39-48), the authors describe two variants of muslin (OT and GW) that seem to be related cases where moslin's alpha parameter is set to 0 and 1, respectively. (This is reflected late in the manuscript where OT is defined as alpha=0 and GW as alpha=1). However, in the context of the earlier discussion for Figure 2b, these terms are not defined and is confusing to the reader. I suggest you move the definition provided on p. 10 line 20 to the earlier discussion on p.7. Also related, calling the case where alpha=0 "OT" seems a bit misleading since moslin itself is described as an optimal transport algorithm variant. Would it not be better to refer to these two cases as "W" and "GW" as originally described on p. 7 lines 24-25?   
2. Suggested word change on p. 14 line 19: "transcriptomically" -> "transcriptionally"  
3. Suggested word change on p. 17 line 59: "build-in" -> "built-in"  
4. On p. 18 lines 22-25, the authors describe that in vitro experiments are amenable to recurring sampling of lineage states; the authors should consider that this is also possible with some in vivo setups, for example sampling from the blood or via needle biopsies.   
  
In summary, this is a well-written manuscript describing a timely and effective analytical approach for scLT data that will be of interest to the community of single-cell genomics researchers. I am enthusiastic for its publication and believe that the suggested revisions will strengthen this study by more clearly presenting these results and better characterizing moslin - both where it excels and fails.

**Reviewer 2**

**Were you able to assess all statistics in the manuscript, including the appropriateness of statistical tests used?**

Yes: Yes, outside of the details we highlight in the review

**Were you able to directly test the methods?**

Yes

**Comments to author:**

Lange, Piran, Klein, and Spanjaard et al. present their manuscript moslin, describing a new approach to predict cell fate through a probabilistic coupling of cells sampled in time series experiments with single-cell lineage tracing information and gene expression. The method relies on an optimal transport framework to map cells probabilistically. The manuscript extends previous work to be able to incorporate lineage information and gene expression across multiple time points to make trajectory and cell fate inferences. Specifically, they adopt a Fused Gromow-Wasserstein objective function to balance lineage relationships and gene expression in their mapping.   
They benchmark their approach against LineageOT, CoSpar, GW, and OT methods, showing that moslin outperforms other tools in the majority of conditions. They provide support for their conclusions through one metric, 'mean error', and otherwise validate their results through simulations and in c.elegans which has a known ground truth. Based on their metric, moslin outperforms other methods like LineageOT and CoSpar and is able to unpack a likely intermediate state in zebrafish development. Moslin also integrates well with existing tools like scverse, being built on moscot and Cell Rank 2 for calculating fate probabilities. In summary, moslin is an improvement over existing tools such as LineageOT that is well integrated with existing tools, aiding adoption. We have some suggestions to improve the manuscript, focused on analysis questions that may be missed or tweeks to the presentation, but overall we're very satisfied with the paper as-is.  
  
Overall Points  
- How important are the marginals and the hyperparameters? What settings do certain hyperparameters work in? What are some recommended strategies for selecting the hyperparameters in the absence of a ground truth? How sensitive is it to different hyperparameters?   
- The paper needs more details about how "ground truth" lineage relationships are established. Often, it is not stated which relationships are being used and how they were established.  
- Moslin seems to work well with both simulated and c.elegans data, demonstrates that it is compatible with Cell Rank 2, and returns accurate results. Could you build in a similar AUC to the Cell Rank paper here given they have ground truth labels from the C.elegans dataset? Do the results hold if you compare a very early versus a late time point? How well does it work if the difference in number of cells is large? Overall methods, it appears the mean error decreases with increasing time. Is this due to cell number? Why is error in 3c calculated over the descendants whereas the remaining error is averaged over the ancestors and descendants?   
- The integration of the Zebrafish development data is nice, leveraging the probability mass concept to show the flow of cell types over the population (though the CIs for that analysis seem high). Is there a way to break the circular training approach here?   
- The stochastic silencing rate ceiling seems low given the dropout in some experimental data sets (> 0.4). Could you enumerate this out to a larger parameter space?  
- Could you expand on the Hamming distance as a measure of barcode distance and diversity? What are the within-and between-average clonal population distances using this metric? It would be helpful to expand on the details and rationale here.   
  
  
Again, this paper is certainly of interest to Genome Biology readers and we'd recommend its acceptance after addressing the concerns detailed above, as well as other reviewer concerns.  
  
Hannah Stuart and Aaron McKenna

**Authors Response**

**Point-by-point responses to the reviewers’ comments:**

We would like to thank all the reviewers for their constructive comments, which have been very helpful for improving our manuscript. Our point-by-point response below outlines extensive new analyses, including the following highlights:

* TedSim simulations: we show that moslin yields accurate predictions in the presence of **emerging cell states**. We also show that moslin is robust to increasing noise levels in lineage barcodes and to data subsampling (**Supplementary Figures 1 and 2**).
* *C. elegans* data: we show that moslin is robust with respect to **errors in lineage tree distances** and to cell numbers. We further show that fate probabilities computed with CellRank 2/moslin are **robust when we change the number of terminal states**, and that moslin achieves the lowest ancestor prediction error for non-ciliated RIM neurons (**Supplementary Figures 5, 6 and 8**).
* On the zebrafish data, we present additional analysis that corroborates the hypothesis that **col11 fibroblasts serve as a reservoir to create col12 fibroblasts (**Response Fig. 6**).**
* Throughout our applications, we added new analyses on the **influence of hyperparameters** on moslin’s performance; we summarize our results in the discussion and provide guidelines for practitioners looking to use moslin **(Supplementary Figures 1, 2, 4 and 18)**.

We have also made numerous improvements to **wording and presentation** in response to reviewer suggestions:

* For our simulations, we now expand on our usage of the hamming distance to measure lineage distances.
* For our *C. elegans* application, we better explain the different data subsets we employed, and we improved our description of fate probability computation.
* We make the zebrafish analysis easier to follow through additional labels, legends and annotations in Fig. 4 and Supplementary Fig. 17. We also included an exemplary annotated lineage tree and we show cell type distributions across time points and individuals (Supplementary Fig. 16). In addition, we better explain how we used a proxy task to tune hyperparameters, avoiding a circular training approach.
* Further, we changed the wording to be more consistent about different variants of Optimal Transport, we added explanations about the notion of ground truth we are using in every data application, and we better explain the metrics we use to score moslin’s predictive performance.

We’ve also updated Fig. 4a to be more distinct from [[1]](https://paperpile.com/c/b4vlGI/lZ8q).

**In the following, we present our response to the reviewer's comments. We restate their original comments (black), give point-by-point answers (green) to the questions, and copy parts of the text or specific panels (blue), which directly correspond to comments or refer to them.**

Reviewer #1

**Summary**

In the manuscript entitled "Mapping lineage-traced cells across time points with Moslin" by Lange\*, Piran\*, Klein\*, Spanjaard\* et al, the authors present a new method for analyzing single-cell lineage tracing (scLT) experiments across multiple time points. This method, termed "moslin", implements an optimal transport (OT) algorithm that regularizes the inference of a coupling matrix across timepoints using lineage information learned from scLT data and paired gene expression data. The authors argue for moslin's effectiveness with both simulated and experimental data, and they propose new putative drivers of C. elegans embryogenesis and biological mechanisms contributing to heart regeneration in a zebrafish model of cardiac injury. In closing, the authors provide a discussion of future extensions to the moslin algorithm, especially how additional modalities may be included.

Overall, the manuscript describes an elegant, timely, and useful approach for analyzing scLT data. The authors' presentation of the algorithm makes a strong case for its improvement over existing approaches, and also demonstrates how the tool interoperates with the ecosystem of single-cell analysis tools in the scverse. I also commend the authors for providing a thorough and well documented python package, complete with an easy-to-follow tutorial. Together, this work will be of broad interest to the single-cell community, especially those working on the analysis of scLT datasets.

While I am generally supportive of its publication, I believe the manuscript should be improved with a few additional simulations, analyses, and clarifications before acceptance.

We thank the reviewer for these kind comments as well as the recommendations, which we address below.

**Major comments**

1. **Depth of simulation experiments.** While the authors make a strong case for moslin's effectiveness over other algorithms via simulations, additional simulation experiments should be used to better characterize moslin's performance. The following experiments should be presented, if possible:

a. A critical step in scLT experiments is inferring phylogenies from barcoding data; while moslin is not a tree inference algorithm, a key question that should be addressed is how errors in tree reconstruction affect moslin's inference. The authors should investigate the relationship between tree reconstruction error (using widely-used statistics like Robinsons-Foulds distance or triplets correct1) and coupling error. The authors can consider scoring the accuracy of tree reconstructions from results presented in Figure 2f. Alternatively, if necessary, to obtain enough variability in tree reconstruction complexity, the authors can

reconstruct trees simulated from a spectrum of scLT parameters (e.g., using NJ on TedSim simulated trees).

We thank the reviewer for this important remark. We decided to study the influence of tree reconstruction errors in the *C. elegans* data, where we have real scRNA-seq data and ground-truth lineage information. We found that, while exact results depend on the pair of time points and the data subset studied, the mean error generally increases by less than 0.1 when we perturb up to 20% of the available lineage information. We included the additional analysis in Supplementary Fig. 5d, and in this letter in Response Fig. 1.

We describe our computations for the added analysis in a new paragraph in the Methods (Section 2.3.1, paragraph “Studying the effect of errors in lineage distances on moslin's performance.”):

We wanted to evaluate how inaccuracies in lineage distance information affect moslin's performance in terms of the mean error. Separately for a set of time-point pairs from the two data subsets (ABpxp and cells with complete lineage information), we used ground-truth lineage information to compute symmetric lineage-distance cost matrices CX and CY, corresponding to cells at early and late time points, respectively. Throughout this analysis, we used optimal moslin hyperparameters as identified in our grid search.

In order to simulate tree reconstruction errors, we perturbed a certain fraction of the information in CX and CY, separately for early and late cells. In particular, given a target percentage c, we extracted the indices corresponding to the upper matrix triangular and picked c% of these indices. Next, we randomly permuted lineage distance information for these c% of indices by sampling without replacement. As some cells might by chance receive the same lineage information through sampling, we computed the actual percentage of permuted cost matrix elements c'%. To maintain cost matrix symmetry, we mirrored the perturbed upper matrix triangular to the lower matrix triangular. We repeated these calculations for cells at early and late time points and averaged their corresponding percentages of actual perturbed matrix elements c'% to arrive at a final measure for the degree of lineage-distance cost matrix perturbation. Separately for each pair of time points and for each data subset, we iterated over target permutation percentages c between 0 and 100%, applied moslin, and recorded the mean error.

We find that some settings, like 450-510 min cells from the ABpxp lineage, are very insensitive to lineage information perturbations, while others, like 170-210 min cells from the ABpxp lineage, are more sensitive to lineage information perturbations, especially once we perturb more than 20% of information. This finding is well aligned with our grid search results, which show that 450-510 min ABpxp cells rely less on lineage information ( α = 0. 9) than 170-210 min ABpxp cells ( α = 0. 98; Response Fig. 7a and Supplementary Fig. 4a). Thus, it appears that in situations where lineage information is very important to reveal cell-state transitions, like for the 170-210 min ABpxp cells, perturbations to lineage information are more detrimental to moslin’s

performance compared to situations where lineage information is less important. We speculate that lineage information is less influential when transcriptome similarity alone is informative about cell state transitions.

We summarize our findings in the main text as follows:

Focusing on moslin’s robustness to inaccuracies in lineage-tree distances, we find that the mean error typically increases by less than 0.1 when we permute up to 20% of non-diagonal cost matrix elements (Supplementary Fig. 5d and Methods).

A comparison of a graph

Description automatically generated with medium confidence

**Response Fig. 1 | | Permuting up to 20% of lineage distances causes small changes in mean error -> Included in our manuscript as part of Supplementary Fig. 5**

Scatter plots, visualizing the dependency of the mean error on the fraction of permuted lineage-distance cost matrix elements for ABpxp cells (left) and cells with complete lineage information (right) across two pairs of time points.

b. A hallmark of developmental systems - as observed in later experimental data analyzed - is the presence of transient states only observed at a given timepoint. It is unclear if the authors adequately capture this scenario in their TedSim simulations presented in Fig 1c-f. The authors should either clarify that their simulations generalize to this scenario, or they should study this phenomenon from two perspectives: first, when certain cell states are not observed in the final sampling; second, when there is a large enough temporal gap between samplings such that multiple cell-type transitions might occur, some of which might be transitional cell states not observed at the final timepoint.

We thank the reviewer for noting the importance of studying and reporting the mapping performance as a function of observed cell states. To address this we have re-analysed the TedSim data, considering a larger temporal time gap between early and late cells which induced an *emergent state* at the later time point, not seen at the early time point. First, we note that within this larger gap, moslin maintains its superiority in minimal mean error when compared to competing methods (Response Fig. 2a; Figure 2f). Next, we assess the accuracy of the mapping of emergent states. For each cell at the later time point we query whether the most probable cell inferred as its ancestor is indeed the ground truth parent. We aggregate this score over cells from emergent and non-emergent (persistent) states for comparison (Response Fig. 2b; Supplementary Fig. 2d). Notably, while depicting a larger variance, moslin’s overall mapping accuracy is consistent across emergent and non-emergent states. We have added this analysis, along with additional robustness tests, to the revised manuscript and updated the text accordingly (see our response to your comment 1d).

Of note, the situation of internal transient states cannot be directly constructed using the TedSim framework, which is, to the best of our knowledge, the most advanced temporal dynamics simulator for scRNA-seq data coupled to lineage information. TedSim defines the cells’ state based on a given *cell state tree*, which is provided as input and defines the possible cell states and the developmental path between them. That is, in the construction of the cell division tree, cell states are assigned based on the reference *cell state tree* (Methods section 2.2: TedSim simulated data). Given this construction, it was not possible to mimic an internal transient state using TedSim. In addition, manually removing a state at a later time point will also not mimic the desired behavior, as it will simply imply that cells from the removed state do not have descendants in the tree.

c. While in the simplified simulation framework convergent evolution is explored, it does not appear to be explored in the more sophisticated scLT-based simulations (Figure 2c-f). The authors should provide a TedSim-based simulation experiment in which an amount of convergent evolution is modeled.

We thank the reviewer for raising this point. Unfortunately, within the TedSim simulation framework, we cannot model convergent evolution. As described in our reply to your comment 1b above, TedSim simulates the cell states based on a cell state tree. The cell state tree models the cell differentiation process to determine the future state when a cell divides asymmetrically. Importantly, the cell state tree is based on a Newick format representing rooted phylogenetic trees and does not allow for convergent states. We note that this as well as the previous remark reflect current limitations of the TedSim framework. While developing a novel simulation model is beyond the scope of this work, we have added to the manuscript a discussion regarding the need for such an extended framework in future works:

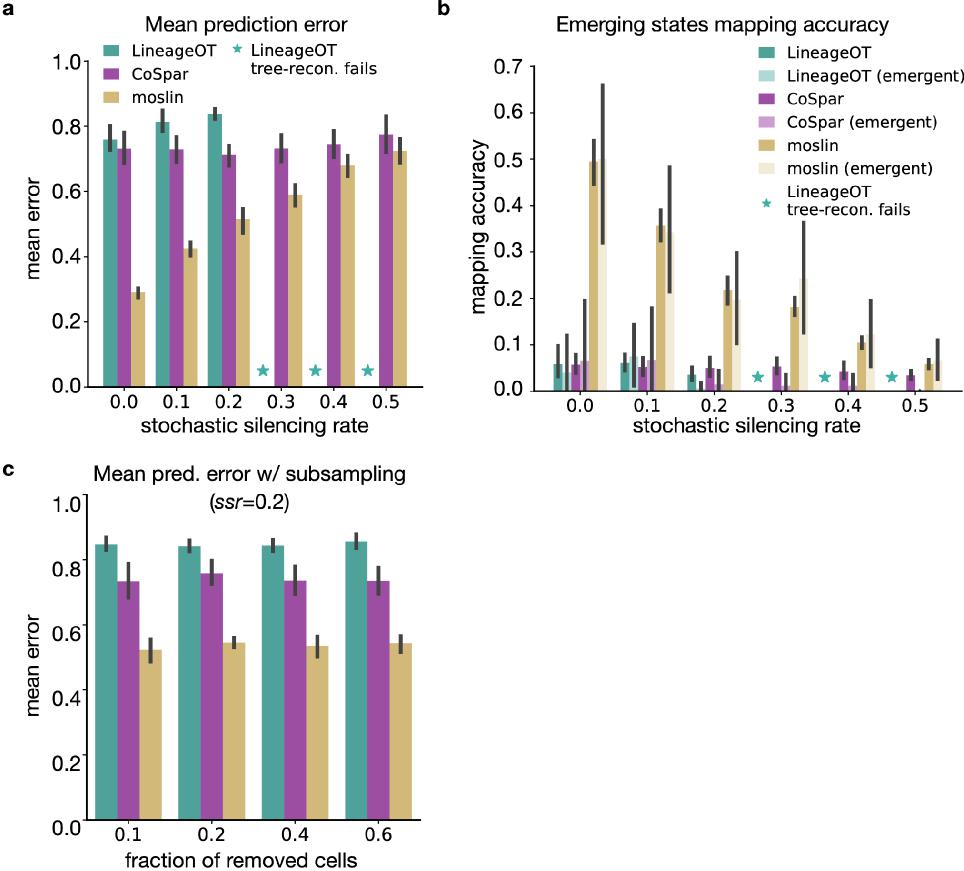
We believe that development of novel simulation frameworks, which are capable of modeling diverse temporal dynamic regimes, will allow major advancements and further validation of suggested analysis tools.

d. Finally, a central challenge to the analysis of scLT data across applications is the bias that may be introduced during sampling. The authors should present a robustness analysis in which error is evaluated against various sampling errors.

Following this remark we have added a robustness analysis of moslin, and competing methods, over a subsampling setting. To preserve the notion of ground truth, we subsampled cells from the late time point, implying that all remaining cells have known ancestors in the tree (Response Fig. 2c; Supplementary Fig. 2e). We observed that all methods are robust to the described subsampling procedure. We would like to thank the reviewer for making this suggestion, as well as previous remarks, which we have included in the manuscript and relate to as follows:

At last, using the ground truth coupling, we validate that moslin faithfully captures transitions to emergent states that appear only at the later time point and that it is robust to subsampling of cells at the later time point (Supplementary Fig. 2e-f, Methods)

**Response Fig. 2 | Evaluation of moslin’s performance over the TedSim simulation-> Included in our manuscript as part of Fig. 2 and Supplementary Fig. 2**



**a.** Mean prediction error of moslin compared to CoSpar and LineageOT, as a function of the stochastic silencing rate. Error bars depict the 95% confidence interval across 10 random simulations. **b**. The mapping accuracy of emergent states, cell states that only appear at the later time point, compared to non-emergent states. For a cell at a late time point an accurate mapping is a mapping in which the most probable ancestor corresponds to its ancestor in the ground truth tree. **c.** The mean prediction error of moslin compared to CoSpar and LineageOT, as a function of the number of removed cells from the late time point. Results are shown for = 0. 2. Error bars depict the 95% confidence interval across 10 random simulations.

2. **Analysis of C. elegans embryogenesis.** While the authors present a thorough analysis of how moslin improves the analysis of C. elegans embryogenesis, the following should be addressed:

a. In the section of text on p10 lines 29-34, the authors should more clarify the difference between C. elegans cells with precise lineage information and those without; above, the authors state they decided to focus on the ABpxp lineage because it is well-annotated. What distinguishes the cells with precise lineage and imprecise lineage in ABpxp, also presented in Supplementary Figure 3c?

We thank the reviewer for bringing our attention to this potentially confusing point. Previously, we had specified the distinction between the different cell sets in more detail in the Methods (Section 2.3 “*C. elegans* embryonic development” -> “Pre-processing”):

To evaluate moslin's performance, we required ground-truth lineage information. The original study's [[2]](https://paperpile.com/c/b4vlGI/QKgm) mapping inferred partial lineage information for a subset of approx. 46k cells. To obtain **complete** lineage information, we implemented two suggestions by [[3]](https://paperpile.com/c/b4vlGI/aDvo):

* **Strategy 1:** subsetting to the ABpxp lineage. This is a symmetric lineage where "x" indicates either the right ("r") or the left ("l") cell.
* **Strategy 2:** subsetting to all cells with **complete** lineage information.

As the lineage for cells obtained from strategy 1 is not fully specified due to "x", the two strategies lead to disjoint subsets of cells, allowing us to test moslin's performance in two different scenarios.

Note that we changed the wording from “precise” to “complete” in an effort to be more clear. In other words, cells from the “ABpxp” lineage do not have complete lineage information because the “x” can stand for either “right” (“r”) or “left” (“l”). Therefore, ABpxp cells and cells with complete lineage information form two disjoint subsets. To make this distinction clearer, we added the following sentence to the main text:

Note that cells from the ABpxp sublineage do not have complete lineage information because “x” can replace either “r” or “l” (Methods).

b. Legend to Figure 3e: the authors should more clearly state what the "fate probabilities" correspond to, and how they are computed.

We thank the reviewer for this comment. In our original submission, we briefly explained “fate probabilities” in the Methods (Section 2.3 “*C. elegans* embryonic development” -> “Identifying terminal states and computing aggregated fate probabilities.”) by writing:

We aggregated individual terminal states to represent Ciliated neurons, Non-Ciliated neurons, and Glia and excretory cells, by combining the 30 cells identified per state. We computed

absorption probabilities on the Markov chain towards these combined cell sets per terminal state group and interpreted these as fate probabilities.

In our revised manuscript, we expanded the explanation of fate probabilities in the Methods by adding:

In other words, for each non-terminal cell, we initialized several random walks and recorded the terminal cell set they reached. Taking the number of random walks to infinity, these 'arrival frequencies' converge to absorption probabilities, which can be computed efficiently in CellRank 2 [[4]](https://paperpile.com/c/b4vlGI/lvKY).

We additionally extended a sentence in the legend of Fig. 3e to point readers to the additional Methods text:

**e.** UMAPs of aggregated fate probabilities towards Ciliated neurons, Non-ciliated neurons, and Glia and excretory cells (Supplementary Fig. 11 and Methods), **computed via absorption probabilities in CellRank 2 (Methods)**.

c. It should be stated more clearly why 7 terminal states were chosen in the analysis of C. elegans embryogenesis - is there any dependence on the number of terminal states and the robustness of the presented results?

We chose 7 terminal states to include some of the most abundant cell states for each of Ciliated neurons (ASH and AWC), Non-ciliated neurons (AVK, RIM and SIB) and Glia and Excretory cells (AMso and Excretory\_gland). However, we found that our results do not strongly depend on the exact number of terminal states; we included the additional analysis in Supplementary Fig. 8, and in this letter in Response Fig. 3.

In particular, we varied the number of terminal states from five to ten and found that mean macrostate time and purity remained high (Response Fig. 3a). When we increased the number of terminal states, we recovered additional Non-ciliated neuronal populations and the mean fate probability of progenitor cells towards non-ciliated neurons increased accordingly (Response Fig. 3a).

We checked to what extent the overall increase in fate probability towards non-ciliated neurons had an effect on the relative probability of individual cells to commit towards either one of the three aggregated terminal states, Ciliated neurons, Non-ciliated neurons, or Glia and excretory cells. To evaluate this effect for a given aggregated terminal state, we computed fate probabilities for varying numbers of macrostates and calculated all vs. all Person correlations (Response Fig. 3b). For all three aggregated terminal states, we found high Pearson correlations (all above 0.96; Response Fig. 3c). These results indicate that, within each aggregated terminal state, the relative probability of each cell to commit towards that state is very robust when we vary the number of terminal states. As downstream analysis, like driver gene computation, relies on correlations of gene expression with fate probabilities towards a

given aggregated terminal state, robustness of fate probabilities implies robustness of downstream analysis. We summarize these new results in our manuscript as follows:

As expected, predicted terminal states mostly consist of late-stage cells, and each only contains cells from a single cell type (Supplementary Fig. 8a). **When we vary the number of terminal states from five to ten, these terminal states continue to consist mostly of late-stage cells from a single cell type (except for one terminal state when we compute 10 terminal states).**

We also find that moslin/CellRank 2 aggregated fate probabilities are robust with respect to changes in the number of terminal states from five to ten (Supplementary Fig. 8).

Note that we do not show results for 9 terminal states because the 9th and 10th eigenvalues (in order of absolute magnitude) of the CellRank 2/moslin-computed transition matrix are complex conjugates, implying that their corresponding Schur vectors span an invariant subspace of the Markov transition matrix only when kept together. Thus, CellRank 2 automatically increases the number of terminal states from 9 to 10 (and warns about this when running the code), to ensure that valid results are returned. This behavior is explained in detail in the method section of the original CellRank 1 manuscript [[5]](https://paperpile.com/c/b4vlGI/4GsF).

|  |  |  |  |
| --- | --- | --- | --- |
| 1.00 |  |  |  |
| 0.99 | 1.00 |  |  |
| 0.98 | 1.00 | 1.00 |  |
| 0.97 | 0.99 | 1.00 | 100 |

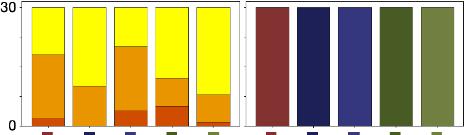
6 7 8 10

a Macrostates

Mean time = 478.00

Macrostate time and purity

Macrostate purity = 1.00



0.09

-0.62

0.29

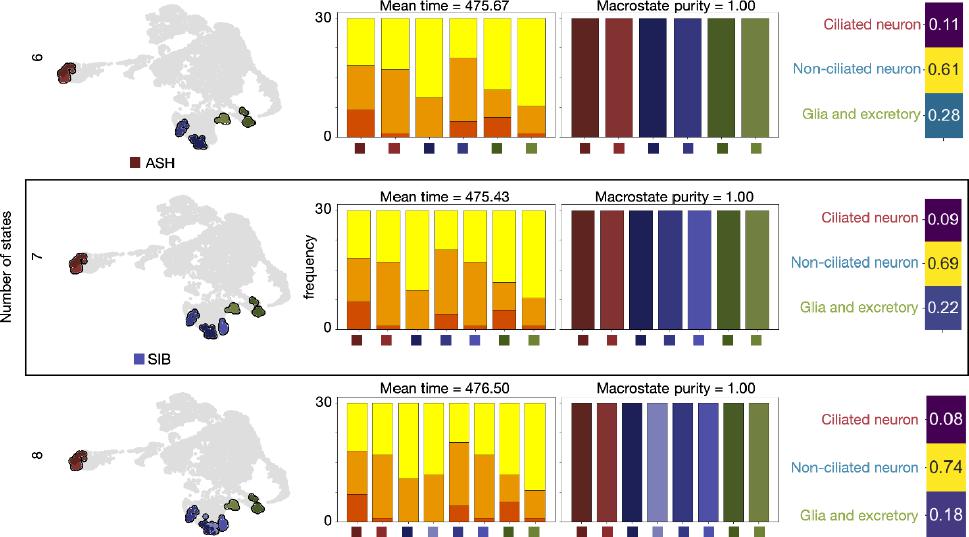
Average fate probability

among progenitors

Ciliated neuron Non-ciliated neuron Glia and excretory

\*et

* AWC • AVK AM▪ so
* RIM Excr. gland



30

Ciliated neuron Non-ciliated neuron Gila and excretory

4

'sit

* ASH

0

Number of states

Ciliated neuron Non-ciliated neuron Gila and excretory

KOK

Mean time = 475.67

Mean time = 476.50

30

II

0.08

0.18

* Other non-ciliated neuron 1

Ciliated neuron Non-ciliated neuron Gila and excretory



Mean time = 468.00 Macrostate purity = 0.95

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | 30 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ***-r*** | | | |  |  |
| 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* Other non-ciliated neuron 2
* Other non-ciliated neuron 3 macrostates

b Comparing fate probabilities C Pearson correlation of fate probabilities for different numbers of macrostates

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2  :cc?, | 5 | example: Non-ciliated neurons | 10 | Ciliated neurons |
| 2 **e. U** |  |  |  |  |

Non-ciliated neurons Gila and excretory

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0:  E | IS1 aggregate and |  |  |  |  |  |  |  |  |  |  |  | |
|  | 47 compute fate | 010 | 1.00 |  |  |  |  | 1 00 |  |  |  | 1 00 | |
| Si' | * probabilities | 0  ro |  |  |  |  |  |  |  |  |  |  | |
| a  3  its n | or - | 16 8  G | 0.99 | 1.00 |  |  |  | 0.99 | 1.00 |  |  | 1.00 | |
| 2  a | t | 2 7 | 0.99 | 100 | 100 |  |  | - 0.98 | 0.99 | 1.00 |  | - 0.99 | |
| 2 I  ***a*** | low 'high | I ° |  |  |  |  |  |  |  |  |  |  | |
|  | correlate 1 all vs. all | 2, 6 |
| 0.98 | 1.00 | 1.00 | 1.00 |  | 0.97 | 0.98 | 1.00 | 1.00 | - 0.98 | |
|  |  | '5 |  |  |  |  |  |  |  |  |  |  | |
|  | 1 0 |  |  |  |  |  |  |  |  |  |  |  |  |
| i | i | z  high $ | 0.98 | 1.00 | 1.00 | 1.00 | 1.00 | 0.96 |  |  | 1.00 | 1 00 | 0.97 |
| I?  8 | I | low | 5 | 6 | 7 | 8 | 10 | 5 | 6 | 7 | 8 | 10 5 | |

Number of macrostates

**Response Fig. 3 | moslin/CellRank 2 terminal states. -> Included in our manuscript as Supplementary Fig. 8.**

**a.** Visualizing terminal states (left), their composition (middle), and average fate probabilities (right) for different numbers of macrostates (rows). UMAPs in the left column show the top 30 cells per moslin/CellRank 2 computed terminal state. For row, we indicate which macrostates emerged in addition to existing macrostates. Bar charts in the middle column show how the top 30 cells per terminal state distribute across time points and clusters. Timepoints and clusters are colored according to [Fig. 3](https://docs.google.com/document/d/10f1OzsOXEIDPTB5wqLnBg09CyNAQrtXQ7EkbJ5EfKAM/edit#fig_celegans). Heatmaps in the left column show average fate probability among progenitor cells. The black box highlights seven terminal states, used throughout this manuscript. **b**. Cartoon, illustrating how we compare fate probabilities towards aggregated terminal states (example: Non-ciliated neurons) by correlating fate probabilities towards that state, computed over varying numbers of terminal states. **c**. Heatmaps, displaying Pearson correlation among fate probabilities towards aggregated terminal states, computed for varying numbers of macrostates as in (**a**), following the procedure outlined in (**b**).

**3. Clarity of Figure 4.**

a. The schematic presented in Figure 4a should be improved with the following: (a) increased resolution of projection and lineage plots on right-hand side; (b) labeling of what the projection axes are; (c) improved annotation of lineage trees.

We thank the reviewer for this comment. We have increased the resolution of Figure 4, in particular of the projection and lineage plots on the right-hand side. We have improved the labeling of the projection axes, and added the following text to the legend:

Right-hand side projections show transcriptomic data over time (top) and a representative lineage tree for each timepoint (bottom).

Furthermore, we have added a new visual with an exemplary annotated lineage tree to improve lineage tree readability (Response Fig. 4, included in our manuscript as part of Supplementary Fig. 16).

A diagram of a cell structure

Description automatically generated with medium confidence

**Response Fig. 4 | Overview of the zebrafish dataset. -> included in our manuscript as part of Supplementary Fig. 16**

**b.** Annotated exemplary lineage tree. This lineage tree represents cell divisions during early stage development as measured by LINNAEUS. Every node represents a cell and the pie chart coloring indicates the eventual cell types that originate from this cell. Since not all divisions are measured, the tree is not necessarily binary. Dashed edges indicate that the temporal placement of the cell (relative to the other cells of the same generation) is unsure. The lineage tree branches show that cells from early divisions give rise to distinct adult cell types.

b. The authors should present additional display items of the dataset analyzed here, for example the distribution of cell types across time points and individuals.

We thank the reviewer for this suggestion. We have added a visual with the cell type distribution across time points and individuals (Response Fig. 5, included in our manuscript as part of Supplementary Figure 16).

A table of guitar chords

Description automatically generated

**Response Fig. 5 | Overview of the zebrafish dataset. -> included in our manuscript as part of Supplementary Fig. 16**

**a**. Cell type frequencies per dataset.

1. The heatmaps in Figure 4b should be improved by (a) increasing the resolution of the figure; (b) annotating the colorbar with an informative legend (similarly for Supplementary Figure 17); (c) annotating the dashed lines with an informative label.

We thank the reviewer for this comment - we have improved Fig. 4b and Supplementary Fig. 17 as suggested.

1. The bubbles in Figure 4g should be annotated with Fibroblast cell type names for clarity. Resolution should also be improved.

We thank the reviewer for this comment - we have annotated the bubbles in Fig. 4g and improved the resolution of Fig. 4.

4. The findings presented in Figure 4 describing a transient col11a1a fibroblast population giving rise to the essential col12a1a fibroblast population is an interesting prediction. Can the authors use the dynamics learned from moslin to speculate on the essentiality of

the transient col11a1 population that emerges at 3dpi? Perhaps a synthetic ablation of this population from the dataset might provide additional support that the col11a1a population is an important (or, essential) reservoir giving rise to the col12a1a population?

We thank the reviewer for this intriguing question. Indeed, since col12a1a fibroblasts are essential to zebrafish heart regeneration, their transient precursors should be essential as well. However, an *in silico* ablation might not give us the insight into the essentiality of the transient fibroblasts that we would hope: this would not change the transcriptomes and lineage relationships of the other cell types. However, a careful analysis of the col11-col12 transition probabilities reveals that over the different dataset combinations, the 3dpi col11a1a fibroblasts act as an important reservoir giving rise to the col12a1a population.

We reasoned that a key defining property of a precursor population is that it will expand to create the full compartment of target cells. If col11a1a fibroblasts are a precursor to col12a1a fibroblasts and there are few col11a1a fibroblasts at 3dpi, they would have to expand strongly. Similarly, to create a high number of col12a1a fibroblasts at 7dpi, the col11a1a fibroblasts at 3dpi would have to expand strongly. In other words: if the col11a1a fibroblasts are a precursor to col12a1a fibroblasts, their expansion would increase as the ratio of 7dpi col12 to 3dpi col11 fibroblasts increases. We calculated both the 3dpi col11 expansion - defined as the mean transition probability between 3dpi col11 and 7dpi col12 fibroblasts - and the ratio between 7dpi col12 and 3dpi col11 fibroblasts for every combination of 3dpi - 7dpi datasets where the number of both fibroblasts is greater than 20. We find a correlation of 0.86 between these two quantities (Response Fig. 6). This corroborates that the 3dpi col11a1a fibroblasts are an important reservoir for the 7dpi col12a1a fibroblasts.

A graph of a cell ratio

Description automatically generated with medium confidence

**Response Fig. 6 | Expansion 3dpi col11 fibroblasts is correlated with col12 7dpi/col11 3dpi ratio**

Scatterplot of the average 3dpi - 7dpi transition probability of col11a1a fibroblasts to col12a1a fibroblasts against the ratio of 7dpi col12a1a fibroblasts to 3dpi col11a1a fibroblasts. Every dot represents a combination between 3dpi and 7dpi datasets with more than 20 col11a1a and col12a1a fibroblasts, respectively.

**Minor comments**

1. In the text describing Figure 2b (p.7 lines 39-48), the authors describe two variants of moslin (OT and GW) that seem to be related cases where moslin's alpha parameter is set to 0 and 1, respectively. (This is reflected late in the manuscript where OT is defined as alpha=0 and GW as alpha=1). However, in the context of the earlier discussion for Figure 2b, these terms are not defined and is confusing to the reader. I suggest you move the definition provided on p. 10 line 20 to the earlier discussion on p.7. Also related, calling the case where alpha=0 "OT" seems a bit misleading since moslin itself is described as an optimal transport algorithm variant. Would it not be better to refer to these two cases as "W" and "GW" as originally described on p. 7 lines 24-25?

We thank the reviewer for this comment. We agree our original naming was confusing and changed it according to the reviewer’s suggestion. Throughout the text, we now refer to the α = 0 case, which solves a Wasserstein problem for just gene expression information, as “W”.

1. Suggested word change on p. 14 line 19: "transcriptomically" -> "transcriptionally"
2. Suggested word change on p. 17 line 59: "build-in" -> "built-in"

We thank the reviewer for these two suggestions and incorporated them into the main text.

1. On p. 18 lines 22-25, the authors describe that in vitro experiments are amenable to recurring sampling of lineage states; the authors should consider that this is also possible with some in vivo setups, for example sampling from the blood or via needle biopsies.

We thank the reviewer for this point. In our original submission, we elaborated on this point in the Methods section (Section 1.2, “In vivo single-cell lineage tracing”):

In clonal resampling (CR), the aim is to observe the same clone (cells sharing the same barcode) across several time points, i.e., for a single phylogenetic tree, we aim to observe some ancestral nodes besides the leaf nodes. As this approach relies on the repeated sampling of clonally related cells, it applies primarily to in-vitro settings [[6–8]](https://paperpile.com/c/b4vlGI/Askl+UGrG+t2YX), in vivo transplantation settings [[6]](https://paperpile.com/c/b4vlGI/Askl), or in vivo regenerative systems like human PBMC and CD34+ samples [[9,10]](https://paperpile.com/c/b4vlGI/5LY3+UctA) or the zebrafish fin [[11]](https://paperpile.com/c/b4vlGI/tVRe).

We extended a sentence to clarify this point in the main text discussion:

In contrast, for in-vitro experiments **and some regenerative or transplantation in-vivo experiments,** cells from the same population can be sampled at different time points, rendering their lineage information directly compatible across time points **(Methods)**.

In summary, this is a well-written manuscript describing a timely and effective analytical approach for scLT data that will be of interest to the community of single-cell genomics researchers. I am enthusiastic for its publication and believe that the suggested revisions will strengthen this study by more clearly presenting these results and better characterizing moslin - both where it excels and fails.

We thank the reviewer again for their constructive comments and suggestions, and positive evaluation of our work. We also believe that the revisions to the manuscript have clarified and strengthened it.

Reviewer #2

**Summary**

Lange, Piran, Klein, and Spanjaard et al. present their manuscript moslin, describing a new approach to predict cell fate through a probabilistic coupling of cells sampled in time series experiments with single-cell lineage tracing information and gene expression. The method relies on an optimal transport framework to map cells probabilistically. The manuscript extends previous work to be able to incorporate lineage information and gene expression across multiple time points to make trajectory and cell fate inferences. Specifically, they adopt a Fused Gromov-Wasserstein objective function to balance lineage relationships and gene expression in their mapping.

They benchmark their approach against LineageOT, CoSpar, GW, and OT methods, showing that moslin outperforms other tools in the majority of conditions. They provide support for their conclusions through one metric, 'mean error', and otherwise validate their results through simulations and in c.elegans which has a known ground truth. Based on their metric, moslin outperforms other methods like LineageOT and CoSpar and is able to unpack a likely intermediate state in zebrafish development. Moslin also integrates well with existing tools like scverse, being built on moscot and Cell Rank 2 for calculating fate probabilities. In summary, moslin is an improvement over existing tools such as LineageOT that is well integrated with existing tools, aiding adoption. We have some suggestions to improve the manuscript, focused on analysis questions that may be missed or tweeks to the presentation, but overall we're very satisfied with the paper as-is.

We thank the reviewer for the careful reading of our manuscript and the insightful comments. **Comments**

1. How important are the marginals and the hyperparameters? What settings do certain hyperparameters work in? What are some recommended strategies for selecting the hyperparameters in the absence of a ground truth? How sensitive is it to different hyperparameters?

We thank the reviewer for this comment. An advantage of moslin is that we can easily interpret the role of each hyperparameter: weights the importance of gene expression compared to lineage information, ε controls the strength of entropic regularization, and a (only used for the zebrafish dataset) determines the level of unbalancedness at the source marginal, i.e. the earlier time point. Importantly, this knowledge provides guidelines for the directionality of adapting these per setting but does not assist in selecting the actual value, which is dataset dependent. Realizing this poses a limitation for practitioners looking to use moslin, we have attempted to address this in the analysis of the zebrafish dataset, where we lack ground truth. Specifically, we suggest setting initial marginals using growth rates and selecting the hyperparameters , ε and a using a grid search, evaluating over a proxy task “cell type

persistency”. We observed strong robustness with respect to all hyperparameters outside extreme values (Supplementary Fig. 18). Following the reviewer's comment, in the revised manuscript, we emphasize this point in the discussion and suggest its applicability to datasets without predefined ground truth. We have added the following text to the discussion:

We have tested moslin in scenarios with and without a known ground truth. An advantage of moslin is that we can easily interpret the role of each hyperparameter: weights the importance of gene expression compared to lineage information, ε controls the strength of entropic regularization, and a (only used for the zebrafish dataset) determines the level of unbalancedness at the source marginal, i.e. the earlier time point. Importantly, this knowledge provides guidelines for the directionality of adapting these per setting but does not assist in selecting the actual value, which is dataset dependent. We have addressed this in the analysis of the zebrafish dataset, where we lack ground truth. Specifically, we suggest setting initial marginals using growth rates and selecting the hyperparameters , ε and using a grid search, evaluating over a proxy task “cell type persistency”. Performance on this proxy task strongly correlates with performance on the actual target task, the transient fibroblast precursor predictions, suggesting the cell type persistence test can be applied to other datasets without a known ground truth.

While hyperparameter optimality criteria vary between datasets, it is illustrative to compare the hyperparameter regimes and robustness. For the 2-gene simulation dataset, the dependence of model performance on hyperparameters can be found in Supplementary Fig. 1c,d. We observe that performance increases for smaller values of ϵ, controlling the amount of entropic regularization. Furthermore, we see that small values for the GW scaling parameter α are optimal, but performance strongly reduces for α = 0 (where we only use gene expression information). We have now also included an analysis on the TedSim simulation data studying the relationship between the stochastic silencing rate (ssr) and the GW scaling parameter, α. We observe that moslin is robust to small perturbations of this parameter, and that, as the ssr increases, the optimal α decreases, as lineage information becomes less accurate (Supplementary Fig. 2c). We have added the following sentence to the manuscript:

As expected, as we increase the ssr, the lineage information becomes less reliable hence the optimal GW scaling parameter, α, decreases.

In the *C. elegans* data (Supplementary Fig. 4, Response Fig. 7), a similar picture emerges for the entropic regularization parameter ϵ: lower values generally increase performance. The deterministic nature of the developmental lineage tree in *c. elegans* means that lineage information encodes cell transitions, and it is therefore not surprising to see that high values for α are optimal. We have added the following sentence to the manuscript:

As expected from the high resolution lineage information in this dataset, we find that moslin performs best for large values of α, reflecting strong influence of the lineage term (Supplementary Fig. 4a).

The hyperparameter robustness analysis for the zebrafish data can be found in Supplementary Fig. 18. Again, we see that lower values of € yield better results, and so do lower values of α. For the zebrafish, low values of α are expected since the cell type persistency test is transcriptome-based, de-emphasizing the lineage information. The added value of nonzero α values becomes apparent on tasks that are not fully transcriptome-based, such as the prediction of the temporal trajectory of transient fibroblasts (Fig. 4d-e). In the zebrafish dataset, we have

an additional hyperparameter controlling the degree of unbalancedness, τ , whose optimal

values are around 0.4 even though the exact value does not have a strong impact on the performance.

From these observations, we can draw some conclusions that hold independent of the dataset analyzed. First of all, the entropic regularization parameter € should be chosen to be small as long as this does not lead to convergence issues or overfitting. The GW scaling parameter α should have higher values if the datasets have a high degree of lineage information, and lower values if the lineage information is of lower resolution. The exact value of the unbalancedness

parameter τ does not seem to be of strong impact on the performance in our analysis;

however, this particular parameter was only used in one dataset. We have added the following text to the manuscript:

Through our analyses, we have identified guiding principles on the choice of hyperparameters: First of all, the entropic regularization parameter € should be chosen small as long as this does not lead to convergence issues or overfitting. The GW scaling parameter α should have higher values if the datasets have a high degree of lineage information, and lower values if the lineage

information is of lower resolution. The exact value of the unbalancedness parameter τ does not

seem to be of strong impact on the performance in our analysis; however, this particular parameter was only used in one dataset.

A screenshot of a graph

Description automatically generated

**Response Fig. 7 | Optimal hyperparameters for moslin on *C. elegans* embryogenesis. -> included in our manuscript as Supplementary Fig. 4**

**a-b.** Heatmaps displaying the mean over ancestor and descendant errors for moslin on cells from the ABpxp sublineage (**a**) and cells with complete lineage information (**b**). Grey color indicates non-converged runs, black boxes highlight optimal hyperparameter combinations.

2. The paper needs more details about how "ground truth" lineage relationships are established. Often, it is not stated which relationships are being used and how they were established.

We thank the reviewer for this comment. Below, we highlight details of “ground truth” lineage relationships in every application and corresponding changes we made to the text.

**Simulations**

In simulations, the ground-truth lineage relationships are known by design. Both simulation setups model cellular division as a binary tree and store this ground-truth lineage tree for method evaluation.

***C. elegans* embryogenesis**

For *C. elegans*, the invariant, ground-truth lineage tree is known. However, the full gene expression profile for every node in this lineage tree is not known. Thus, previous work sequenced approx. 86,000 *C. elegans* cells and mapped their gene expression profiles to *C. elegans* lineage tree nodes, using the expression of known marker genes and imaging of fluorescent reporter genes [[2]](https://paperpile.com/c/b4vlGI/QKgm). In our original submission, we described this process in the main text as follows:

This species' ground truth lineage tree is known [[12]](https://paperpile.com/c/b4vlGI/n8kSd) and available to assess moslin’s reconstruction performance.

Previous work mapped time-series gene expression profiles of approx. 86k single cells to individual tree-nodes [[2]](https://paperpile.com/c/b4vlGI/QKgm), providing a setting where joint lineage, state and time information is available.

To be more explicit about the nature of ground truth in this dataset, we added the following sentence to the main text:

We treat the original author’s mapping [[2]](https://paperpile.com/c/b4vlGI/QKgm) of cells to the *C. elegans* lineage tree as ground truth and use it to evaluate our algorithm.

**Zebrafish heart regeneration**

Comprehensive ground-truth lineage relationships across time points for this setting are unknown. However, as a proxy to ground truth, we assume that most persistent cell states should be their own ancestors. In other words, we compute a coupling between early and late cells, and we check whether persistent cell states are coupled correctly to themselves. In our original submission, we described this evaluation in the main text as follows:

Initially, we validate the performance of moslin and other methods in this challenging regeneration setting. We design a test around the assumption that most persistent cell states should be their own precursor; for example, precursors of atrial endocardial cells at 7dpi should be atrial endocardial cells at 3dpi. We used this test to select optimal hyperparameter values.

Briefly, for each cell at t2 and for each method, we select the most probable t1 ancestor and calculate the accuracy between the t2 and t1 cell types, jointly for cells at 3 and 7 dpi (Fig. 4b and Methods).

Thus, while we do not have access to ground-truth lineage relationships for each and every cell, we use prior knowledge about this biological system to evaluate whether a persistent subset of cell types is correctly mapped to themselves. We now make this more explicit in the main text:

Thus, while ground truth lineage relationships across time points are unknown in this setting, we initially restrict our attention to a subset of persistent cell states, which we assume to evolve into themselves. This methodology is applicable in human and model organisms such as mouse and zebrafish, where lineage relationships are not deterministic as they are in *c. elegans*.

We treat this evaluation as a pretext task which allows us to tune algorithm hyperparameters. Using the set of hyperparameters identified in this pretext task, we then continue to evaluate algorithm predictions for transient fibroblast states. By tuning hyperparameters on a pretext task where some notion of ground truth is available, we aim to avoid the circular training approach the reviewer alluded to below in comment 4.

3. Moslin seems to work well with both simulated and c.elegans data, demonstrates that it is compatible with Cell Rank 2, and returns accurate results. Could you build in a similar AUC to the Cell Rank paper ==here given they have ground truth labels from the C.elegans dataset? Do the results hold if you compare a very early versus a late time point? How well does it work if the difference in number of cells is large? Overall methods, it appears the mean error decreases with increasing time. Is this due to cell number? Why is error in 3c calculated over the descendants whereas the remaining error is averaged over the ancestors and descendants?

We thank the reviewer for these suggestions. We split the comment into four parts and go through them one by one.

**Computing AUC values**

In the CellRank v1 paper [[5]](https://paperpile.com/c/b4vlGI/4GsF), we considered an in-vitro reprogramming process of mouse embryonic fibroblasts towards induced endoderm progenitor cells [[7]](https://paperpile.com/c/b4vlGI/UGrG) with two possible outcomes: successful reprogramming or reprogramming failure. To quantify CellRank’s success in predicting reprogramming outcome for cells at early days, we framed the situation as a binary classification problem and evaluated accuracy by plotting the receiver operating characteristic (ROC) curve and computing the associated area under the ROC curve (AUC).

On the C. elegans data, we have additional information that allows us to evaluate accuracy more precisely. In particular, we have access to ground-truth lineage information at the single cell level. Moslin’s coupling matrix P probabilistically relates cells in early and late time points. For example, for early cell i, row i in matrix P contains the likelihood that cell i will transition into any cell j at the later time point. We could use these values to train a multi-class classifier that

predicts the most likely descendant, and evaluate using AUC values. However, this metric would not take into account how *close* we were to the true descendants in gene expression. In contrast, the descendant error we employ in moslin compares the predicted descendant distribution for cell with the ground-truth distribution, and scores the discrepancy in terms of the Wasserstein distance. Crucially, this distance takes the geometry of the underlying phenotypic landscape into account. By repeating these calculations of every early cell , we obtain the average descendant error. We proceed analogously for late cells via the average descendant error. The mean over average descendant and ancestor errors is the reported “mean error”, which quantifies algorithm performance precisely. Quantifying algorithm performance in this way was suggested in [[3]](https://paperpile.com/c/b4vlGI/aDvo) and renders our results directly comparable to those of LineageOT.

We now emphasize the geometric properties of our error metrics in the Methods (Section 2.1):

Crucially, this distance takes the geometry of the underlying phenotypic landscape into account. Couplings to cells that are not the actual ancestors or descendants of the reference cell incur a larger penalty in mean error if they are further away from the true ancestors or descendants in terms of their gene expression states.

Thus, given the finer lineage resolution in the C. elegans data, we believe that the mean over ancestor and descendant errors is a more suitable (and previously suggested [[3]](https://paperpile.com/c/b4vlGI/aDvo)) metric to quantify and compare algorithm performance.

**Linking a very early to a very late time point directly**

We evaluated this question by coupling cells over increasing time gaps. Specifically, for the ABpxp lineage, we directly mapped 170 min cells to all other time points. We fixed hyperparameters ( α = 0. 95 and ϵ = 0. 01; compare with Response Fig. 7a) and computed the mean error between moslin-predicted and ground truth couplings. As expected, we found that the mean error for these larger time gaps was greater than the pairwise mean error for all time points (Response Fig. 8).

Larger time gaps usually imply greater shifts in cell-state distributions, which render computational trajectory reconstruction more challenging for any computational method, including moslin. We speculate that the mean error does not increase monotonically with increasing time gaps because the fixed hyperparameters we choose might be more suitable for some time point-pairs than for others (Response Fig. 7 and Supplementary Fig. 4).

A graph of different colored bars

Description automatically generated

**Response Fig. 8 | Mapping accuracy as a function of the time gap**

Bar charts over the mean error between predicted and ground-truth cell couplings for adjacent time points (left) and between 170 min cells and all other time points (right). On the right, we show results for moslin with fixed hyperparameters.

**Dependency on cell numbers**

We investigated algorithm performance as a function of cell number and included the additional analysis as part of a new Supplementary Fig. 5, and in this letter in Response Fig. 9.

The difference in the number of cells is largest on the 330/390 min pair of time points, for both ABpxp cells and cells with complete lineage information (Response Fig. 9c). For both data subsets, moslin on this pair of time points outperforms all competing algorithms, except for GW on cells with complete lineage information when initialized with the W solution (Response Fig. 9a,b). In particular, moslin’s performance on 330/390 min cells is better than moslin’s performance on some other time points, which have smaller cell number differences. Thus, we conclude that cell number differences are not a main factor determining moslin’s performance.

We agree with the reviewer that moslin’s mean error decreases over time for ABpxp cells (Response Fig. 9a). However, this pattern is different on cells with complete lineage information, and we could not detect a clear link to the number of cells per time point (Response Fig. 9b). We hypothesize that moslin’s performance depends on more complex factors, like the distance in phenotypic space between cell populations at two adjacent time points, chosen hyperparameters, or the mapping of cells to *C. elegans* tree nodes in the original publication [[2]](https://paperpile.com/c/b4vlGI/QKgm).

We summarize the additional analysis in the main text as follows:

The number of cells per time point and the difference in cell number between adjacent time points are not main factors determining moslin’s performance ([Supplementary Fig.](https://docs.google.com/document/d/10f1OzsOXEIDPTB5wqLnBg09CyNAQrtXQ7EkbJ5EfKAM/edit#sup_celegans_full) 5b,c).

A graph of different numbers

Description automatically generated with medium confidence

**Response Fig. 9 | Benchmark results and cell numbers -> Included in our manuscript as part of Supplementary Fig. 5**

**a.** Bar chart of the mean error for LineageOT [[3]](https://paperpile.com/c/b4vlGI/aDvo), W, GW, and moslin across time points for ABpxp cells (left) and cells with complete lineage information (right). For cells with complete lineage information, the red box on the 330/390 min time point pair shows GW’s performance when initialized with the W solution; GW in that case slightly outperforms moslin in terms of the mean error (0.11 to 0.12). **b.** Bar chart over the number of cells per time point, grouped by data subset. **c.** Bar chart over the difference in cell number between two adjacent time points, grouped by data subset.

**Ancestor error for Fig. 3c**

In panel 3c, our aim was to zoom in onto one specific population, given by the RIM\_parent cells, and to highlight their predicted descendants by different methods. In our revised manuscript, we also computed the ancestor error over RIM cells and added it to the corresponding Supplementary Fig. 6. We highlight the relevant panels below in Response Fig. 10, where we show that moslin achieves the lowest ancestor error.

A graph of a graph showing the different types of graphs

Description automatically generated with medium confidence

**Response Fig. 10 | Ancestor and descendant error on the 330/390 min time point pair. -> Included in our manuscript as part of Supplementary Fig. 6**

**c.** Ground truth (leftmost) and predicted (right) 390 min descendants of 330 min RIM\_parent cells by different methods. Black lines indicate coupling matrix elements. “error” indicates the aggregated descendant error for 330 min RIM\_parent cells, as in [Fig. 3](https://docs.google.com/document/d/10f1OzsOXEIDPTB5wqLnBg09CyNAQrtXQ7EkbJ5EfKAM/edit#fig_celegans). **d.** Ground truth (leftmost) and predicted (right) 330 min ancestors of 390 min RIM cells by different methods. 390 min cells are colored in light gray, 330 min cells are colored according to their ancestor likelihood. Pie charts visualize the aggregated ancestor distribution over the clusters shown in (**a**), using the color scheme of (**a**). **“error” indicates the aggregated ancestor error over 390 min RIM cells.**

4. The integration of the Zebrafish development data is nice, leveraging the probability mass concept to show the flow of cell types over the population (though the CIs for that analysis seem high). Is there a way to break the circular training approach here?

We thank the reviewer for the compliment and the question. As with all machine learning approaches, we agree it is important that moslin is able to make predictions that it is not trained on. In the absence of single cell-level ground truth to use for training, we have selected the

concept of cell-type persistence: for any cell type A that is present at t\_1 and t\_2, cells of type A at t\_2 should mostly originate from cells of that same type A at t\_1. This means that the question we are investigating for the zebrafish about the role of transient cell types is not trained for, making the answer a real prediction. We have changed the main text to reflect this:

To test moslin’s ability to predict complex temporal relationships between cell types outside its training regime of persistent cell types, we evaluated its performance in a setting where cell types are not identical between time points.

Incidentally, we agree with the reviewer that the confidence intervals for transient fibroblast transitions are surprisingly high. We speculate that this is due to low and strongly variable numbers of the relevant cell types: the experimental design was unbiased. To answer this specific question to a higher degree of precision, datasets that are enriched for fibroblasts would have definitely helped.

1. The stochastic silencing rate ceiling seems low given the dropout in some experimental data sets (> 0.4). Could you enumerate this out to a larger parameter space?

We thank the reviewer for making this comment. In the updated manuscript we have indeed extended the analysis to stochastic silencing rate of ssr = 0. 5. Notably, moslin retains superior performance also in this parameter regime (Figure 2f). In addition, we now elaborate on the *stochastic\_silencing\_rate* procedure in the methods section:

We add to the TedSim simulated barcodes a stochastic silencing rate, corresponding to the rate at which entire segments (cassettes) are removed from the barcode. In the TedSim simulation, each cassette has 4 characters and there are 8 cassettes per barcode.

Finally, we would like to highlight that we borrowed the *stochastic\_silencing\_rate* regime from Cassiopeia’s [[13]](https://paperpile.com/c/b4vlGI/cgtz) simulation pipeline. Looking for valid parameter regimes, we relied on reports in the original publication [[13]](https://paperpile.com/c/b4vlGI/cgtz) as well as methods for tree reconstruction which rely on it [[14]](https://paperpile.com/c/b4vlGI/MNq5). In these publications, the maximal value considered was ssr = 0. 2. An additional validation to the viability of this range is the observation that LineageOT struggles with tree reconstruction at ssr > 0. 2 (the reconstruction algorithm fails due to completely null barcodes).

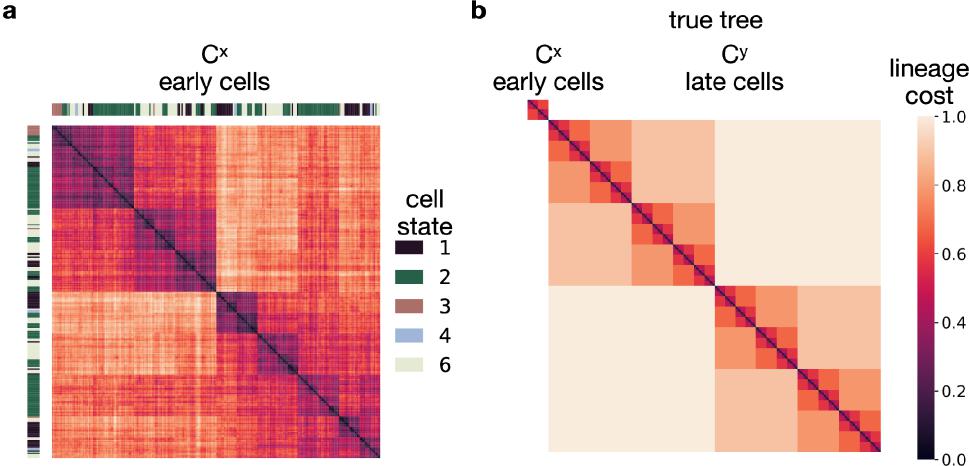
1. Could you expand on the Hamming distance as a measure of barcode distance and diversity? What are the within-and between-average clonal population distances using this metric? It would be helpful to expand on the details and rationale here.

We would like to first relate to the rationale of this approach–we reasoned that it is valuable to decouple moslin’s performance from the tree reconstruction accuracy of available methods. Furthermore, the hamming distance lifts the need for tree reconstruction. This allows working in regimes where tree reconstruction algorithms may fail, as we observe for ssr > 0. 2 in LineageOT’s application to the TedSim simulated data.

Importantly, the evolving lineage tracing regime lacks the notion of discrete clones, and the lineage based distances are intended to capture *local* distances within the lineage tree. That is, we would like the cost to increase as distance between tree leaves increases. The complementary expression distance matrix shall capture cell state relations.

We see that the Hamming distance approach indeed manages to capture the local tree structure, presenting the similarity of neighboring cells (Response Fig. 11a, Supplementary Figure 2a). That is, the distance matrix between cells (tree leaves) sorted according to their appearance in the ground-truth tree depicts the expected block wise structure, as observed in the tree based distances used for the 2D setting (Response Fig. 11b, Supplementary Fig. 1a).

**Response Fig. 11 | Lineage distance cost matrices reflect ground truth lineage**



**similarity. -> Included in our manuscript as part of Supplementary Figures 1 and 2**

Heatmaps, visualizing the lineage distance cost matrices in the simulated datasets TedSim (**a**) and 2-gene bifurcation trajectory (**b**). Cells are sorted according to their order of appearance in the true tree. **a.** Hamming distance-based lineage cost between early cells in a TedSim simulation (for clarity of visualization, only early cells are shown) **b.** Tree distance based lineage cost between early and late cells, respectively, as used by moslin in the bifurcation trajectory setting of the 2-gene simulation for the true tree.

Again, this paper is certainly of interest to Genome Biology readers and we'd recommend its acceptance after addressing the concerns detailed above, as well as other reviewer concerns. Hannah Stuart and Aaron McKenna

We thank the reviewers again for their positive evaluation of our work and for their constructive comments, which helped us improve the manuscript.

References

1. [Hu B, Lelek S, Spanjaard B, El-Sammak H, Simões MG, Mintcheva J, et al. Origin and function of activated fibroblast states during zebrafish heart regeneration. Nat Genet. 2022;54: 1227–1237.](http://paperpile.com/b/b4vlGI/lZ8q)
2. [Packer JS, Zhu Q, Huynh C, Sivaramakrishnan P, Preston E, Dueck H, et al. A lineage-resolved molecular atlas of C. elegans embryogenesis at single-cell resolution. Science. 2019;365. doi:](http://paperpile.com/b/b4vlGI/QKgm)[10.1126/science.aax1971](http://dx.doi.org/10.1126/science.aax1971)
3. [Forrow A, Schiebinger G. LineageOT is a unified framework for lineage tracing and trajectory inference. Nat Commun. 2021;12: 4940.](http://paperpile.com/b/b4vlGI/aDvo)
4. [Weiler P, Lange M, Klein M, Pe’er D, Theis F. CellRank 2: unified fate mapping in multiview single-cell data. Nat Methods. 2024. doi:](http://paperpile.com/b/b4vlGI/lvKY)[10.1038/s41592-024-02303-9](http://dx.doi.org/10.1038/s41592-024-02303-9)
5. [Lange M, Bergen V, Klein M, Setty M, Reuter B, Bakhti M, et al. CellRank for directed single-cell fate mapping. Nat Methods. 2022;19: 159–170.](http://paperpile.com/b/b4vlGI/4GsF)
6. [Weinreb C, Rodriguez-Fraticelli A, Camargo FD, Klein AM. Lineage tracing on transcriptional landscapes links state to fate during differentiation. Science. 2020;367. doi:](http://paperpile.com/b/b4vlGI/Askl)[10.1126/science.aaw3381](http://dx.doi.org/10.1126/science.aaw3381)
7. [Biddy BA, Kong W, Kamimoto K, Guo C, Waye SE, Sun T, et al. Single-cell mapping of lineage and identity in direct reprogramming. Nature. 2018;564: 219–224.](http://paperpile.com/b/b4vlGI/UGrG)
8. [Hurley K, Ding J, Villacorta-Martin C, Herriges MJ, Jacob A, Vedaie M, et al. Reconstructed Single-Cell Fate Trajectories Define Lineage Plasticity Windows during Differentiation of Human PSC-Derived Distal Lung Progenitors. Cell Stem Cell. 2020;26: 593–608.e8.](http://paperpile.com/b/b4vlGI/t2YX)
9. [Penter L, Gohil SH, Lareau C, Ludwig LS, Parry EM, Huang T, et al. Longitudinal single-cell dynamics of chromatin accessibility and mitochondrial mutations in chronic lymphocytic leukemia mirror disease history. Cancer Discov. 2021. doi:](http://paperpile.com/b/b4vlGI/5LY3)[10.1158/2159-8290.CD-21-0276](http://dx.doi.org/10.1158/2159-8290.CD-21-0276)
10. [Lareau CA, Ludwig LS, Muus C, Gohil SH, Zhao T, Chiang Z, et al. Massively parallel single-cell mitochondrial DNA genotyping and chromatin profiling. Nat Biotechnol. 2021;39: 451–461.](http://paperpile.com/b/b4vlGI/UctA)
11. [Alemany A, Florescu M, Baron CS, Peterson-Maduro J, van Oudenaarden A. Whole-organism clone tracing using single-cell sequencing. Nature. 2018;556: 108–112.](http://paperpile.com/b/b4vlGI/tVRe)
12. [Sulston JE, Schierenberg E, White JG, Thomson JN. The embryonic cell lineage of the nematode Caenorhabditis elegans. Dev Biol. 1983;100: 64–119.](http://paperpile.com/b/b4vlGI/n8kSd)
13. [Jones MG, Khodaverdian A, Quinn JJ, Chan MM, Hussmann JA, Wang R, et al. Inference of single-cell phylogenies from lineage tracing data using Cassiopeia. Genome Biol. 2020;21: 92.](http://paperpile.com/b/b4vlGI/cgtz)
14. [Sashittal P, Schmidt H, Chan M, Raphael BJ. Startle: A star homoplasy approach for CRISPR-Cas9 lineage tracing. Cell Syst. 2023;14: 1113–1121.e9.](http://paperpile.com/b/b4vlGI/MNq5)

**Second round of review**

**Reviewer 1**

I would like to commend the authors of the manuscript entitled "Mapping lineage-traced cells across time points with Moslin" on formulating a comprehensive response and strong revision. They have addressed all of my concerns and I have no additional suggestions. I am in favor for its acceptance and believe this will be of great interest to the field of researchers using single-cell lineage tracing approaches.

**Reviewer 2**

Looks good, thanks for addressing our concerns!