Supplemental information

Benchmarked approaches for reconstruction of

*in vitro* cell lineages and *in silico* models of

*C. elegans* and *M. musculus* developmental trees

Supplemental Information

Figure S1. *Mus musculus in silico challenge (Related to Figure 1)* A. Simulation of the Mouse lineage, “token” cells whose lineage are stochastically chosen to be followed as the lineage tree is formed, are shown in blue, in white are represented cells whose lineage is not followed. At the end of the simulation for the mouse lineage information for about 10,000 blue cells is stored, but it is estimated that the size of the tree is about $10^{12}$ or a trillion cells. B. Visualization of the 10,000 cell Mouse tree with 11 types of cells encoded by different colors.

Figure S2. *In vitro challenge rankings for all teams according to multiple metrics (Related to Figure 2).*
The ranks for each team were evaluated by calculating the ranksum values (left boxplots) for the Robison-Foulds (middle boxplots) and the triplet metric (right boxplots) sampled 1000 times with replacement from the scores for the 30 individual trees. The 9 teams were ordered by average ranksum and the Bayes Factor (BF) was calculated, yellow boxes show teams that are considered to be tied as they have a $1/3 < BF < 3$ and a $BF > 3$ against all the other teams in grey. Implementation of a third metric calculating quartets could not differentiate the top 3 teams: Yosef Lab (*Cassiopeia*) 0.4200, Guan Lab 0.4232, Jasper06 0.4243.

Figure S3. *In vitro challenge results with Robinson-Foulds and triplets metrics (Related to Figure 2).*
The participant teams’ distribution of scores across 30 reconstructed lineage trees is shown for A. triplets metric B. Robinson-Foulds metric C. Histogram showing the difference between the Robinson-Foulds and triplets metrics for all 30 trees across all teams. Median of zero indicates that overall the metrics agree but dispersion suggests a small bias for higher distance values in triplets. D. The histogram of scores of all 30 trees for all 9 teams are for left Robinson-Foulds and right triplets metrics, color coded depending on the percentage of unique barcode arrays in the tree. Deep blue dots trees with 25-50% unique arrays, gray blue dots trees with 50-75% unique arrays, light blue dots, trees with 75-100% unique arrays. E. Comparison of team performance depending on whether cells with degenerate barcodes are merged (gold boxes) or not (blue boxes). Left Boxplots represent the triplet distances, Right RF distances, of trees where
for both predictions and ground truth, cells with the same barcodes were merged into a single leaf. The procedure followed for each tree a 100x bootstrap choosing each time a different cell with the same barcode as distances were recalculated for each fold.

**Figure S4.** *In vitro* challenge list of trees that were reconstructed perfectly by at least one team (Related to Figure 2). Ground truth lineages are shown along with the array state for each cell.

**Figure S5.** *In vitro* challenge largest trees with high reconstruction scores (Related to Figure 2).

Two examples of large trees with 29 and 23 cells respectively and their RF and Triplets distance. These large but accurate trees were reconstructed by A) *AMberLand* and B) *Yosef Lab* (*Cassiopeia*).

**Figure S6.** *Guan Lab* approach for *in vitro* challenge (Related to Figure 3) A. Probability of mutations for the array sites and their corresponding weights for the Hamming distance. When calculating the weights for the Hamming distance, the mutation direction preference is set as reciprocal of the mutation frequency so that the rarer the mutation type, the more weight it is given to the distance between cells. B. A rule-based Hierarchical clustering approach was used to generate the trees. The cells character arrays final states were transformed by weights according to the observed probability of mutations, and the transformed states were used to calculate the distance between cells. The hierarchical clustering was done using a rule-based method to reconstruct parent cells, based on the fact that the editions from initial states (1) to edited states (0 and 2) are irreversible. C. Comparison of different clustering methods for the distance matrices including Rule-based hierarchical clustering, UPGMA and Neighbor Joining. The performance is shown for both triplets and RF distances. The Distribution across the 30 lineages in the test set and the average of the two tree measurements is shown by the violin plots. The rule-based hierarchical clustering method and UPGMA have similar performance on reconstructing cell lineage trees.

**Figure S7.** Representation of the decision tree and weights (Related to Figure 5) obtained by *Amberland* using GBM for the training set in the *in vitro* challenge. For each decision tree leaf
are indicated: on top the feature’s weight, the number of cells \( n \) and the percentage of the training set cells they represent, and in bold is the criteria of the feature used for selecting the next leaf i.e. number of times the feature is present when comparing the 2 cells character arrays. Features in this case are: F1-both not mutated, F2-both same mutation F3-one mutation F4-different mutations. This figure was made using the R package “rattle”.

**Figure S8. Reconstructing trees by clustering probability matrices as implemented by AMbeRland for the training set of the in vitro challenge (Related to Figure 5).** Seventy six trees of different number of cells were used to optimize the tree reconstruction thresholds from the probability matrix of cells being sisters obtained from training a GBM algorithm A. Performance of the algorithm for four sets of thresholds: set_A=(0,0,0,0,0) results in mean RF=0.512 and triplets=0.389; set_B=(0.5,0,0,0,0) results in mean RF=0.519 and triplets=0.380; set_C=(0.8,0.4,0.2,0.1,0.05) results in mean RF=0.512 and triplets=0.433; and set_D=(0.3,0.1,0.05,0.01,0.005) results in mean RF=0.502 and triplets=0.375. The numbers shown in the scatter plots represent the tree ID and the color represents the number of cells in the tree. Threshold set_D was used to reconstruct the test dataset for submission. B. A perfectly reconstructed tree with 3 thresholds (tree ID 70 from the training set, RF=0 and triplets=0) has 7 pairs joined into clusters at level 1, 4 pairs joined at level 2 and 2 pairs joined at level 3. C. Probability matrices for tree 70 are plotted for each level. From here it can be seen that cells 7 and 8 have the highest probability so they are first joined into cluster C1, the next pair with highest probability comprises cells 12 and 13 which joined into cluster C2 and so on. Once all pairs are defined, the algorithm moves to Level 2, where clusters C2 and C3 have the highest pairwise probability (cells on these two clusters can be seen on top right corner of level 1 probability matrix) so they are joined into a new cluster C1. The algorithm proceeds until all cells are joined into a single lineage.

**Figure S9. Clustering of cells into trees performed by AMbeRland for the training set in the C. elegans in silico challenge (Related to Figure 5).** One hundred trees of a hundred cells each were used to optimize the tree reconstruction thresholds from the probability matrix of cells being sisters obtained from training a GBM algorithm A. Comparing performance of the algorithm for two sets of thresholds: set_A={0} gives mean RF distance=0.78 and triplets=0.59;
set_B=(0.07, 0.04, 0.01, 0.05, 0, 0, 0,0) gives mean RF distance=0.71 and triplets=0.49.
Threshold set_B was used to reconstruct the test sample. B. Ground truth and reconstructed tree for training sample 100, with RF distance = 0.48 and triplets=0.44. C. Probability matrices for training sample 100 are plotted for each level. Clusters identified letters C. by Four clusters for level 7 (C1-C4) are indicated on the reconstructed tree in B.

Figure S10. Agreement distribution across all reconstructed trees at different normalized tree depths for the in vitro challenge (Related to Figure 6). A depth of 0 represents the root of the tree whereas a depth of 1 corresponds to the leaves and therefore the depth of cell divisions within the lineage fall between [0,1]. Top For a given ground truth lineage, The Felsenstein Bootstrap Support is calculated across all reconstructed trees submitted by the teams corresponding to that lineage. We obtain a distribution by computing the FBS score for all 30 ground truth lineages. Bottom The Transfer Bootstrap Expectation is calculated in an analogous way.
Figure S3
Figure S4 Trees that at least one team inferred perfectly
A  Tree 29, number of cells = 29
Ground truth

Inference: AMberLand
RF = 0.44
Triplets = 0.40

B  Tree 28, number of cells = 23
Ground truth

Inference: Yosef Lab
RF = 0.48
Triplets = 0.70

Figure S5
Figure S7
Figure S10