HERACLES to the Rescue: Hyperbolic Embeddings for Reconstruction and Analysis of CRISPR-Cas9 LineagES

Gilad Turok*
Department of Computer Science
Columbia University
New York, New York 10025
gt2453@columbia.edu

Abstract

Tracking the fate of single cells is immensely important in contemporary biology. With the advent of CRISPR-Cas9 genome editing and high-throughout single-cell sequencing, this is now a reality. In this paper, we present HERACLES, a novel phylogeny-reconstruction algorithm to reconstruct the cell lineage of CRISPR-Cas9 experiments. HERACLES finding hyperbolic embeddings that accurately represent the phylogenetic tree of interest while addressing the unique challenges of CRISPR-based lineage tracking. We provide early simulation experiments that demonstrate the efficacy of HERACLES, surpassing the current state of the art algorithms.

1 Introduction

The ability to track individual cells throughout biological processes has significant and immense potential, especially during regimes of cellular development or cancer. With the advent of CRISPR-Cas9 gene editing and single-cell sequencing technologies, we can now track the lineage of a particular cell by inserting a heritable *casette* into its genome. Each casette contains multiple *target sites* on which CRISPR-induced insertions and deletions (*indels*) can occur and be passed down to its descendents. Many generations later, these casettes are sequenced and the indels at each target site are recorded, forming a *character matrix*. With this character matrix, a phylogenetic algorithm attempts to reconstruct the original cell-lineage as a phylogenetic tree as in 1. Using this method, biologists have discovered unprecented insights into zebrafish McKenna et al. [2016] Raj et al. [2018] and mouse development Kalhor et al. [2018] Chan et al. [2019], among other contexts.

Many phylogeny-reconstrution algorithms, however, are not well-suited for cell-lineage data because of its (1) scale, (2) sequencing challenges, and (3) unique evolutionary model. As such, traditional algorithms – such as Neighbor Joining Saitou and Nei [1987], Weighbor Bruno et al. [2000], and Camin-Sokal Camin and Sokal [1965] – may not perform well in this domain. These algorithms are suited for a few samples, not tens of thousands of cells. This largely makes them computationally intractable for cell-lineage data. Furthermore, these algorithms are unable to handle the vast amounts of missing data that occur in single-cell-lineage tracing: missing data may be *heritable* – from large-scale CRISPR-induced deletions – or *stochastic* – from incomplete capture of target sites during sequencing. Finally, these algorithms do not take into account the design of lineage tracers and the evolutionary stochastic models they induce. Unlike cannonical evolutionary models, lineage tracing experiments are time-irreversible such that once an edit is made, reversions to an unedited start or a different mutation are impossible. Furthermore, in this setting the ancestral state, or root of the

*https://gil2rok.github.io/

Preprint. Under review.

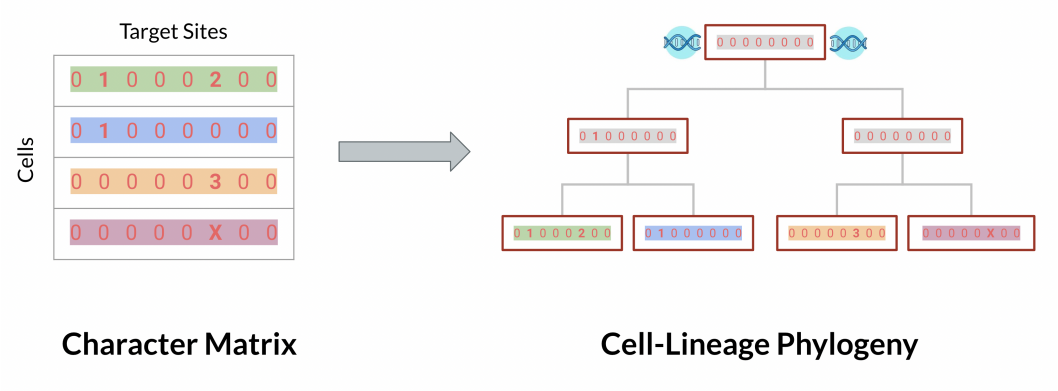

Figure 1: From a character matrix, HERACLES seeks to reconstruct the original cell-lineage as a phylogenetic tree.

lineage tree, is known to be the unedited state; this information can be used to inform the construction of the phylogenetic tree. Thirdly, lineage tracing systems contain site-specific mutation rates that enable capturing evolutionary relationships at varying time-scales.

We aim to tackle these challenges by developing a novel phylogeny-reconstruction algorithm HERACLES: Hyperbolic Embeddings for Reconstruction and Analysis of CRISPR-Cas9 LineagES. HERACLES explicitly considers (1) different mutation rates at each target site, (2) non-uniform distributions over indels at each target site, and (3) time-irreversible evolutionary model of lineage tracing experiments. HERACLES broadly works by finding hyperbolic embeddings that accurately represent the phylogenetic tree of interest. First we identify the continous time Markov-chain that underlies the evolutionary process of the lineage tracing experiment. In this context, we construct a log-likelihood function that quantifies the fidelity of a candidate phylogenetic tree. Next, we embed the character matrix into hyperbolic space and use gradient-based optimization to find the embedding that maximizes the log-likelihood function. Finally, we reconstruct the phylogenetic tree from these hyperbolic embeddings. We provide early simulation experiments that demonstrate the efficacy of HERACLES.

2 Related Work

Previous work has been carried out in Wilson [2021] that reconstructs phylogentic trees with hyperbolic embeddings for homologous sequence data, generated under the Jukes-Cantor evolutionary model. They find that using hyperbolic embeddings outperforms many traditional phylogeny-reconstruction algorithms in a variety of simulated settings. This paper (similarly) reconstructs phylogenetic trees with hyperbolic embeddings but does so for the evolutionary model of CRISPR-Cas9 lineage tracing. Furthermore, Jones et al. [2020] is the only work that directly tackles lineage tracing for CRISPR-Cas9. They propose three approaches: greedily optimizing for parsimony, finding a Steiner Tree with Integer Linear Programming (ILP), and a hybrid approach that combines the two. HERACLES can be viewed as a combination of these two works, learning hyperbolic embeddings for CRISPR-Cas9 lineage tracing experiments.

3 Methods

3.1 Preliminaries

Character Matrix Every row in the character matrix C describes the casette for one of N cells and each casette contains a variety of target sites. A target site σ in cell c can be unedited, mutated, or deleted – although a variety of different mutations exists. Formally, let $\sigma_c \in \{0, 1, \ldots, M_\sigma, D\}$ where 0 represents the unedited state, $1, \ldots, M_\sigma$ represent the different mutations, and D represents a deletion.

Transition Matrix The accumulation of CRISPR-Cas9 induced edits at a given target site σ evolves as a time-homogeneous continuous time Markov chain (CTMC). This emits a row-stochastic matrix P^σ that defines a time-homogeneous and time-irreversible model of mutation. This transition matrix P^σ is target-site dependent but we will drop the explicit dependence on target site σ and just write P. For mutation states a, b, the entry $P_{ab}(t)$ is the probability of transitioning from state a to state b in t time steps.

Infinitesimal Generator As a CTMC, the probablity matrix P is determined by its infinitesimal generator Q by

$$P_{ab}(t) = e^{t \cdot Q_{ab}} \tag{1}$$

For each target site σ, the time to leave the unedited state is exponentially distributed with rate $\lambda = \lambda_{M\sigma} + \lambda_D$ for the site-specific rate $\lambda_{M\sigma}$ and the (small) deletion rate λ_D. Given that a transition occurs to some other character state, the transition to the observed character state, s, occurs with probability $p_{\phi_\sigma s}$ with $\sum_{i \in M_\sigma} p_{\phi_\sigma i} = 1$. (Note P represents the transition matrix while p represents the indel distribution.)

The site-specific mutation rate $\lambda_{M\sigma}$, deletion rate λ_D, and indel distribution p can be determined by sensor screens performed a priori or inferred directly from the character matrix.

A toy example of the infinitesimal generator Q is found in the appendix 5.

Log-likelihood Function Consider σ_i and σ_j, a specific target site σ for cells i and j respectively. Because CRISPR-Cas9 lineage tracing is time-irreversible, we must consider a common ancestor a and examine the time it takes from a to evolve into states σ_i and σ_j. We make the simplifying assumption that the ancestral state a is equidistant to both states σ_i and σ_j.

Thus we compute the likelihood of the two cells being separated by an evolutionary distance of t time units as:

$$\mathcal{L}_{ij}^P(t) = \prod_{\text{site } \sigma} \sum_{a \in A(\sigma_i, \sigma_j)} \pi_a \; P_{a\sigma_i}(t/2) \; P_{a\sigma_j}(t/2). \tag{2}$$

π_a is the prior probability of being in an ancestral state a, $P_{a\sigma_i}$ is the probability of transitioning from a to σ_i in $t/2$ time units, and $P_{a\sigma_j}$ is the probability of transitioning from a to σ_j in $t/2$ time units. Furthermore, we iterate over all target sites σ and over all feasible ancestors $A(\sigma_i, \sigma_j)$ for cells i and j at site σ.

The ancestral priors π_a are learned from Felstenstein's algorithm Felsenstein [1973]. The feasible ancestors $A(\sigma_i, \sigma_j)$ are constrainted to a limited set of states, including the unedited state 0 and the least common ancestor of σ_i and σ_j. We compute the set of feasible ancestors as

$$A(\sigma_i, \sigma_j) = \begin{cases} \{0, \sigma_i\} & \text{if } \sigma_i = \sigma_j \\ \{0\} & \text{if } \sigma_i \neq \sigma_j \end{cases} \tag{3}$$

In practice we compute the log-likelihood to avoid numerical issues. Thus the log-likelihood function ultimately quanitifes the fidelity of cell i and cell j being separated by an evolutionary distance of t time units, as informed by our assumptions about the transition matrix P.

3.2 The HERACLES Algorithm

The HERACLES algorithm consists of three steps: embedding the character matrix into hyperbolic space, optimizing the hyperbolic embeddings with respect to the log-likelihood function, and recovering a phylogenetic tree from the optimized hyperbolic embeddings. HERACLES is presented in its entirety in 1 and we now describe each step in detail.

Embedding the Character Matrix into Hyperbolic Space To embed the character matrix C into hyperbolic space, we first compute a pairwise distance matrix D with the Hamming distance

Algorithm 1 HERACLES

Require: Character matrix C, Transition matrix P

D =HammingDist(C) ▷ Pairwise distance matrix
T =TreeReconstruction(D) ▷ Reconstructed tree
X =SarkarsConstruction(T) ▷ Initial hyperbolic embedding

while not converged **do**
  for $i = 1, \ldots, N$ **do**
    $L = \sum_j \log \mathcal{L}_{ij}^P \left(d(x_i, x_j)\right)$ ▷ Log-likelihood for cell i
    $x_i \leftarrow$ RiemmanianAdam$(x_i, \nabla_{x_i} L)$ ▷ Gradient Update
  end for
end while

D' =GeodesicDist(X) ▷ Pairwise distance matrix
T' =TreeReconstruction(X) ▷ Reconstructed tree

between each pair of cells. We then use a tree reconstruction algorithm – such as Neighbor Joining – to construct an intial phylogenetic tree T. Finally, we use a generalization of Sarkar's construction Sarkar [2012] Sala et al. [2018] to isometrically embed the tree into hyperbolic space as a point cloud, serving as the initial embedding X to be optimized.

Optimizing the Hyperbolic Embeddings We optimize the hyperbolic embeddings iteratively for one cell-embedding at a time. For each cell i, we sum over a minibatch of cells j and compute the log-likelihood:

$$L = \sum_j \log \mathcal{L}_{ij}^P \left(d(x_i, x_j)\right) \tag{4}$$

$$= \sum_j \log \left[\prod_{\text{sites } \sigma} \sum_{a \in A(\sigma_i, \sigma_j)} \pi_a \; P_{a\sigma_i}\left(\frac{d(x_i, x_j)}{2}\right) \; P_{a\sigma_j}\left(\frac{d(x_i, x_j)}{2}\right) \right]. \tag{5}$$

Note that we set $t = d(x_i, x_j)$ such that instead of σ_i and σ_j being seperated by t time units, we use the geodesic distance $d(x_i, x_j)$ between the embeddings x_i and x_j.

After computing the log-likelihood, we update the embedding x_i of cell i with the Riemannian Adam optimizer Bécigneul and Ganea [2018]. Riemannian Adam is a modified Adam optimizer Kingma and Ba [2014] that ensures the update remains on the manifold of hyperbolic space. After iterating through all N cells, we get an updated embedding X. This process is repeated until some convergence criteria is met.

Recovering the Phylogenetic Tree Once the hyperbolic embeddings X have converged, we compute a pairwise distance matrix D' with the geodesic distance between each pair of cells. We then use a tree reconstruction algorithm – such as Neighbor Joining – to construct a phylogenetic tree T'. The reconstructed tree of cell-lineages, T', is the final output of the HERACLES algorithm.

4 Experiments and Discussion

To test HERACLES, we simulated a ground truth phylogeny with a birth death process and then overlayed casettes on top of this tree topology. We can then compare the true tree to an estimated tree with the triplets correct metric Sand et al. [2013] – the higher the better. In these experiments, we investigate the performance of HERACLES, the Cassiopeia greedy algorithm and the Cassiopeia ILP algorithm.

In small simulations of $N = 20$ cells, HERACLES' learns good hyperbolic embeddings: the log-likelihood and triplets correct metric both increase. Furthermore, HERACLES outperforms both state-of-the-art Cassiopeia algorithms.

	HERACLES	Cassiopeia ILP	Cassiopeia Greedy
20 cells	**0.465**	0.457	0.310
1000 cells	**0.260**	0.049	0.10

Table 1: Results of HERACLES, Cassiopeia ILP and Cassiopeia Greedy on simulated data.

In larger simulations of $N = 1000$ cells, HERACLES again outperforms both state-of-the-art Cassiopeia algorithms. However, in this setting, HERACLES is unable to improve the initial hyperbolic embeddings as the log-likelihood and triplets correct metric fluctuate stochastically but do not increase. This suggests that the initial embedding scheme into hyperbolic space works well but the gradient updates from RiemannianAdam do not.

5 Conclusion

In this paper we explored reconstructing cell-lineages of CRISP-Cas9 experiments. We introduced HERACLES, a novel algorithm that embeds a character matrix into hyperbolic space, improves the hyperbolic embeddings with RiemannianAdam, and then reconstructs the original phylogenetic tree. We showed that HERACLES outperforms state-of-the-art Cassiopeia algorithms on simulated data.

In future work, we plan to investigate HERACLES' performance in a much wider variety of simulated settings to ensure robust performance. We also plan to explore alternative embedding schemes for the initial embedding into hyperbolic space. Furthermore, we plan to extensively explore why HERACLES fails to improve the hyperbolic embeddings in larger simulations. Finally, we plan to apply HERACLES to real CRISP-Cas9 data.

References

Aaron McKenna, Gregory M. Findlay, James A. Gagnon, Marshall S. Horwitz, Alexander F. Schier, and Jay Shendure. Whole-organism lineage tracing by combinatorial and cumulative genome editing. *Science*, 353(6298), July 2016. doi: 10.1126/science.aaf7907. URL `https://doi.org/10.1126/science.aaf7907`.

Bushra Raj, Daniel E Wagner, Aaron McKenna, Shristi Pandey, Allon M Klein, Jay Shendure, James A Gagnon, and Alexander F Schier. Simultaneous single-cell profiling of lineages and cell types in the vertebrate brain. *Nature Biotechnology*, 36(5):442–450, March 2018. doi: 10.1038/nbt.4103. URL `https://doi.org/10.1038/nbt.4103`.

Reza Kalhor, Kian Kalhor, Leo Mejia, Kathleen Leeper, Amanda Graveline, Prashant Mali, and George M. Church. Developmental barcoding of whole mouse via homing crispr. *Science*, 361 (6405):eaat9804, 2018. doi: 10.1126/science.aat9804. URL `https://www.science.org/doi/abs/10.1126/science.aat9804`.

Michelle M Chan, Zachary D Smith, Stefanie Grosswendt, Helene Kretzmer, Thomas M Norman, Britt Adamson, Marco Jost, Jeffrey J Quinn, Dian Yang, Matthew G Jones, et al. Molecular recording of mammalian embryogenesis. *Nature*, 570(7759):77–82, 2019.

Naruya Saitou and Masatoshi Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425, 1987.

William J. Bruno, Nicholas D. Socci, and Aaron L. Halpern. Weighted Neighbor Joining: A Likelihood-Based Approach to Distance-Based Phylogeny Reconstruction. *Molecular Biology and Evolution*, 17(1):189–197, 01 2000. ISSN 0737-4038. doi: 10.1093/oxfordjournals.molbev.a026231. URL `https://doi.org/10.1093/oxfordjournals.molbev.a026231`.

Joseph H Camin and Robert R Sokal. A method for deducing branching sequences in phylogeny. *Evolution*, pages 311–326, 1965.

Benjamin Wilson. Learning phylogenetic trees as hyperbolic point configurations, 2021.

Matthew G Jones, Alex Khodaverdian, Jeffrey J Quinn, Michelle M Chan, Jeffrey A Hussmann, Robert Wang, Chenling Xu, Jonathan S Weissman, and Nir Yosef. Inference of single-cell phylogenies from lineage tracing data using cassiopeia. *Genome biology*, 21(1):1–27, 2020.

Joseph Felsenstein. Maximum likelihood and minimum-steps methods for estimating evolutionary trees from data on discrete characters. *Systematic Biology*, 22(3):240–249, 1973.

Rik Sarkar. Low distortion delaunay embedding of trees in hyperbolic plane. In *Graph Drawing: 19th International Symposium, GD 2011, Eindhoven, The Netherlands, September 21-23, 2011, Revised Selected Papers 19*, pages 355–366. Springer, 2012.

Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *International conference on machine learning*, pages 4460–4469. PMLR, 2018.

Gary Bécigneul and Octavian-Eugen Ganea. Riemannian adaptive optimization methods. *arXiv preprint arXiv:1810.00760*, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Andreas Sand, Gerth Stølting Brodal, Rolf Fagerberg, Christian NS Pedersen, and Thomas Mailund. A practical o (n log 2 n) time algorithm for computing the triplet distance on binary trees. In *BMC bioinformatics*, volume 14, pages 1–9. Springer, 2013.

Appendix

Infinitesimal Generator Q for Toy Lineage Tracing Experiment

	ϕ_1	ϕ_2	1	2	3	D
ϕ_1	$-(\lambda_{M1}+\lambda_D)$	0	$\lambda_{M1}P_{\phi_1 1}$	$\lambda_{M1}P_{\phi_1 2}$	$\lambda_{M1}P_{\phi_1 3}$	λ_D
ϕ_2	0	$-(\lambda_{M2}+\lambda_D)$	$\lambda_{M2}P_{\phi_2 1}$	$\lambda_{M2}P_{\phi_2 2}$	$\lambda_{M2}P_{\phi_2 3}$	λ_D
1	0	0	$-\lambda_D$	0	0	λ_D
2	0	0	0	$-\lambda_D$	0	λ_D
3	0	0	0	0	$-\lambda_D$	λ_D
D	0	0	0	0	0	0

Figure 2: Infinitesimal generator Q of toy lineage tracing experiment. This system containts two target sites σ_1, σ_2 and three potential mutations $1, 2, 3$. The elements of the infinitesimal generator Q are comprised of the site-specific mutation rates $\lambda_{M\sigma}$, the deletion rate λ_D, and the indel distrbution $p_{\phi_\sigma s}$. More specifically, $p_{\phi_\sigma s}$ represents the probability transitioning to the observed character state s from some other state ϕ_σ, conditioned on a transition occuring.