Emulating Dark Matter Halo Merger Trees with Graph Generative Models

Tri Nguyen [1 2 3] **Chirag Modi** [4] **Siddharth Mishra-Sharma** [3 5] **L. Y. Aaron Yung** [6] **Rachel Somerville** [7]

Abstract

Merger trees track the hierarchical assembly of dark matter halos across cosmic time and are crucial for semi-analytic models of galaxy formation. However, traditional methods for generating merger trees rely on ad-hoc assumptions and struggle to incorporate environmental information. We present `FLORAH-Tree`, a generative model for merger trees using directed acyclic graphs, combining recurrent neural networks to capture the sequential nature of mergers with normalizing flows to model halo property distributions. Our model, trained on cosmological simulations, accurately reproduces key merger tree statistics across wide mass and redshift ranges. This approach offers a computationally efficient alternative to simulation-based merger trees while maintaining statistical fidelity, and can be generalized to a broad class of tree-generative problems beyond galaxy formation.

1. Background

In Lambda Cold Dark Matter, dark matter (DM) halos form hierarchically, i.e. massive halos form from the successive merger of smaller, less massive halos (e.g. White & Rees, 1978). Since galaxies form and evolve within massive DM halos, the accretion and merger histories of DM halos are intrinsically connected to the properties of their galaxies.

The hierarchical growth of DM halos is captured by "merger trees", tree-like data structures that link progenitor halos to their descendants across cosmic times. Merger trees are a key ingredient of semi-analytic models (SAMs), which use simplified prescriptions of baryonic physics to map (semi-)observable galaxy properties onto *DM-only merger trees* (e.g. Yung et al., 2019; Hutter et al., 2021). SAMs are a leading approach in modern galaxy formation, as they require significantly fewer computational resources than full hydrodynamic simulations (see Somerville & Davé 2015).

To generate DM-only merger trees, two methods are traditionally employed: (1) extracting them from DM-only (or N-body) simulations and (2) constructing them analytically using the Extended Press-Schechter (EPS) formalism (Press & Schechter, 1974; Bower, 1991). Simulations are more accurate but have high computational costs, which limits the halo mass range that can be captured. EPS-based methods are faster but cannot capture environmental correlations and long-term temporal dependencies (i.e. how a halo's assembly history affects its future evolution) that influence the merger histories (Somerville et al., 2000).

Machine learning offers promising approaches for generating fast merger trees with environmental information, yet this application remains unexplored. Prior studies such as Jespersen et al. (2022); Chuang et al. (2024) focus on mapping halo properties to baryon properties given existing DM halo merger trees, rather than generating the merger trees themselves. Tang & Ting (2022) presented a graph generative model that maps properties of a halo at $z = 0$ to its progenitor graphs at $z = 2$, which could in principle be adopted to generate full merger trees. However, this approach was only demonstrated for these two redshifts, leaving unclear whether it can generate full merger trees across multiple epochs.

Only Robles et al. (2022) has previously employed an image-based generative adversarial network (GAN) to emulate full merger trees, but GANs are susceptible to mode-collapse and require more extensive validation than presented. Their image-based approach also limits generalizability to different temporal and redshift ranges and cannot enforce specific physical constraints, e.g., prohibiting arbitrary splitting of halos. Graph-based models (Liu et al., 2019; 2023) are better suited for merger tree structure but face significant limitations: they require predetermined graph sizes, making them inflexible for the variable branching structure of

[1]Center for Interdisciplinary Exploration and Research in Astrophysics, Evanston, IL, USA [2]NSF-Simons AI Institute for the Sky, Chicago, IL, USA [3]NSF AI Institute for Artificial Intelligence and Fundamental Interactions, Cambridge, MA 02139, USA [4]Center for Cosmology and Particle Physics, New York University, New York, NY, USA [5]Department of Physics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA [6]Space Telescope Science Institute, Baltimore, MD, USA [7]Center for Computational Astrophysics, Flatiron Institute, New York, NY, USA. Correspondence to: Tri Nguyen <trivtnguyen@northwestern.edu>.

Proceedings of the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

merger trees, and they lack mechanisms to enforce the same physical constraints.

Instead of emulating full merger trees all at once, a simpler approach is to first model the main progenitor branch, which traces the most massive progenitors of a halo, and then reconstruct the remaining branches in a physically consistent manner. Following this strategy, Nguyen et al. (2024) developed `FLORAH`, a generative model based on recurrent neural networks (RNNs) and normalizing flows that autoregressively models main progenitor evolution. The study demonstrated that `FLORAH` successfully reproduces population statistics of mass and concentration histories in cosmological N-body simulations, as well as clustering properties and assembly bias for main progenitor branches.

In this work, we extend `FLORAH` to generate complete merger trees by incorporating two key components: a classifier to predict the number of progenitors at each time step, and additional neural network modules to model the mass evolution of multiple progenitors simultaneously. Our approach draws inspiration from autoregressive graph generative models like GraphRNN (You et al., 2018), but is specifically tailored for hierarchical tree structures like the merger histories. We refer to the updated model as `FLORAH-Tree`.

We validate our approach by demonstrating that the generated merger trees accurately reproduce key statistical properties, such as progenitor mass distributions and merger rates. To provide additional context for our model's performance relative to established semi-analytic methods, we further compare our results against merger trees generated using the EPS-based algorithm from Parkinson et al. (2008) as implemented in `SatGen` (Jiang et al., 2021). Lastly, we apply the Santa Cruz SAM to the generated merger trees and predict key galaxy-halo scaling relations such as supermassive black hole mass-halo mass and stellar-to-halo mass relations.

Simulation

We train `FLORAH-Tree` on merger trees extracted from the Very Small MultiDark Planck DM-only simulation (VSMDPL; Klypin et al. 2016). The simulation evolves 3840^3 DM particles, each with a mass of $8.9 \times 10^6\,\mathrm{M}_\odot$, within a cubic volume of $(160\,\mathrm{Mpc}\,h^{-1})^3$, using Planck 2013 cosmology (Planck Collaboration, 2014). We also present results using an alternative training simulation in Appendix B.

We extract DM halos using ROCKSTAR and reconstruct merger trees using CONSITENTTREE (Behroozi et al., 2012a;b). We exclude low-resolution halos that ROCKSTAR does not reliably identify. In particular, we exclude all merger trees whose "root" halos fewer than 500 particles ($4.5 \times 10^7\,\mathrm{M}_\odot$) and all progenitor halos fewer than 200 particles ($1.7 \times 10^6\,\mathrm{M}_\odot$), along with their progenitors.

For the feature of each halo, we use its virial mass, $M_{200\mathrm{c}}$ and DM concentration, $c_{200\mathrm{c}}$ (Navarro et al., 1996). The DM concentration correlates with formation time and large-scale environment (e.g., van den Bosch, 2002; Wechsler et al., 2006; Correa et al., 2015) and plays a crucial role in understanding secondary biases in large-scale structures, such as halo assembly bias. We use the cosmological redshifts, z, of the simulation snapshots as our timesteps.

We extract $300,000$ merger trees from a $(72\,\mathrm{Mpc}\,h^{-1})^3$ sub-volume of VSMDPL and split them $80-20$ between training and validation. For testing, we extract $50,000$ merger trees from an independent $(40\,\mathrm{Mpc}\,h^{-1})^3$ sub-volume.

Methodology

1.1. Forward model

Fig. 1 shows the schematic of `FLORAH-Tree`. We model the assembly and merger histories of the halo *backward in time*, i.e. from low to high redshift. By convention, "descendant" halos (denoted with subscript d) refer to halos at lower redshifts that form from the merger of "progenitor" halos (denoted with subscript p) at higher redshifts.

Given a descendant halo with a feature vector, $\mathbf{x}_\mathrm{d}$, at redshift z_d, our goal is to model the feature vectors of N_p progenitors, $\mathcal{X}_\mathrm{p} \equiv \{\mathbf{x}_{\mathrm{p},1} \ldots \mathbf{x}_{\mathrm{p},N_\mathrm{p}}\}$, at some redshift $z_\mathrm{p} > z_\mathrm{d}$. In this manner, given a root halo at present, we can iteratively generate its complete merger tree by successively predicting progenitor populations at higher redshifts until reaching a predetermined redshift or mass threshold.

To model the progenitors $\mathcal{X}_\mathrm{p}$, we also want to incorporate the descendant's *assembly history*, defined as the sequence of halos tracing back from the descendant to the root. The key observation here is that in the direction of increasing redshift, descendant halos split into progenitor halos that evolve (backward in time) independently without recombining. As such, each descendant halo possesses a *unique* assembly history that influences its progenitor population.

We define a history vector from root halo $\mathbf{x}_\mathrm{r}$ to descendant halo $\mathbf{x}_\mathrm{d}$ as $\mathcal{H}_\mathrm{hist} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N_\mathrm{hist})}\}$, where $\mathbf{x}^{(1)} = \mathbf{x}_\mathrm{r}$ and $\mathbf{x}^{(N_\mathrm{hist})} = \mathbf{x}_\mathrm{d}$, with intermediate halos ordered by increasing redshift. We also track the corresponding redshift history $\mathcal{Z}_\mathrm{hist}$, which is provided as input during generation.

We thus write the conditional probability distribution (PDF) of the progenitor halos as $p(\mathcal{X}_\mathrm{p}, N_\mathrm{p} | \mathcal{H}_\mathrm{hist}, \mathcal{Z}_\mathrm{hist}, z_\mathrm{p})$, where we have absorbed $\mathbf{x}_\mathrm{d}$ and z_d into $\mathcal{H}_\mathrm{hist}$ and $\mathcal{Z}_\mathrm{hist}$, respectively. We then split this distribution into two components:

$$p(\mathcal{X}_\mathrm{p}, N_\mathrm{p} \mid \mathcal{C}) = p_\mathrm{mult}(N_\mathrm{p} \mid \mathcal{C})\, p_\mathrm{prop}(\mathcal{X}_\mathrm{p} \mid \mathcal{C}, N_\mathrm{p}), \quad (1)$$

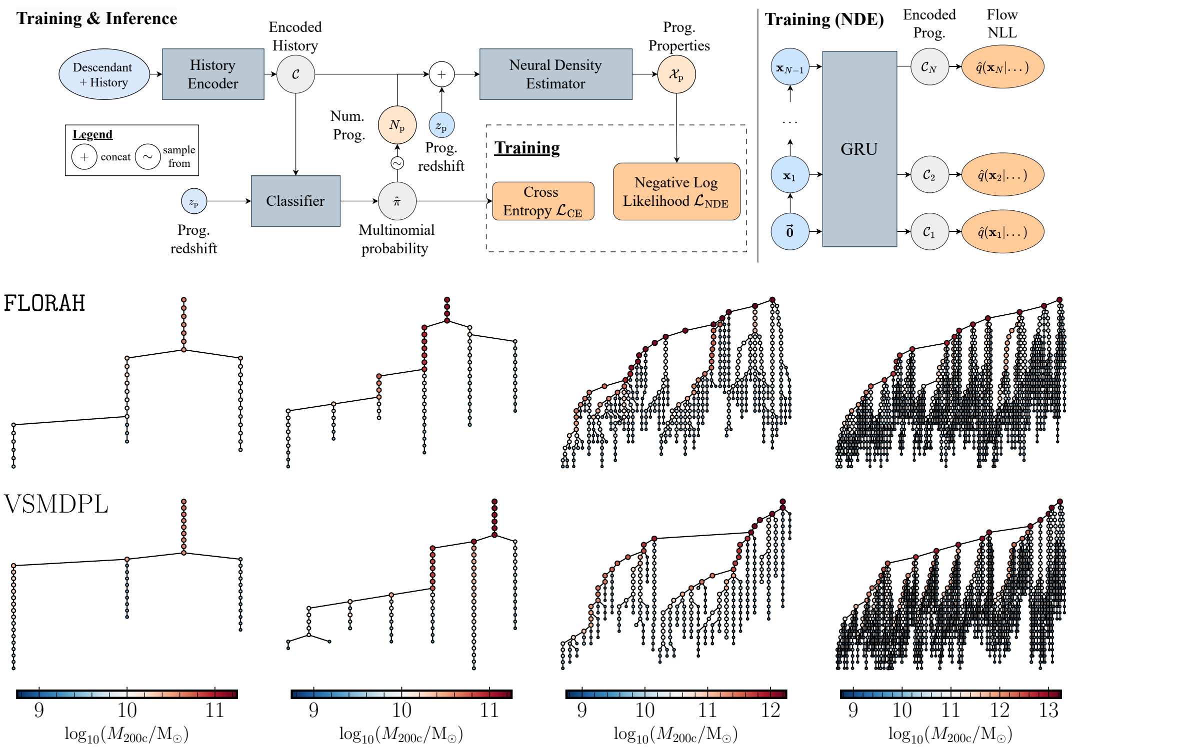

Figure 1. Top: Schematic of `FLORAH-Tree`. *Bottom*: Example `FLORAH-Tree` and VSMDPL merger trees with four different root masses. Vertical positions indicate redshift, increasing from top to bottom, while horizontal positions are arbitrary and do not carry physical meaning.

where $\mathcal{C} \equiv (\mathcal{H}_{\text{hist}}, \mathcal{Z}_{\text{hist}}, z_{\text{p}})$ is the conditioning vector. Here, p_{mult} describes the probability of N_{p} progenitors, and p_{prop} is the joint distribution of progenitor properties.

Assuming a max progenitor count, we can model p_{mult} as a multinomial distribution and employ an FC classifier. Given the timesteps of VSMDPL, we find N_{p} to be typically small, with most mergers involving only $2-3$ massive progenitors. We set the max progenitor count to be $N_{\max} = 3$ and the min progenitor-descendant mass ratio to be 0.01.

The `FLORAH-Tree` model then consists of three primary components: (1) an RNN history encoder to encode $\mathcal{H}_{\text{hist}}$ and $\mathcal{Z}_{\text{hist}}$, (2) an FC classifier to model p_{mult}, and (3) an NDE to model p_{prop}. Below we provide a brief description of the architecture (more details in Appendix A.1).

The encoder consists of 4 GRU layers (Cho et al., 2014) that take in $\mathcal{H}_{\text{hist}}$ and $\mathcal{Z}_{\text{hist}}$ and embed them into a 128-D latent space. Only the output vector from the final timestep in the sequence is used as the encoded representation. Before passing through the GRU, $\mathcal{H}_{\text{hist}}$ and $\mathcal{Z}_{\text{hist}}$ are first independently projected through FC layers, then summed together.

The FC classifier takes in this encoded representation (concatenated with the progenitor redshift z_{p}) and outputs a multinomial probability vector, $\hat{\boldsymbol{\pi}} \equiv (\hat{\pi}_1, \hat{\pi}_2, \ldots, \hat{\pi}_N)$. Here $\hat{\pi}_i$ denotes the probability of having i progenitors.

Since the progenitor count varies per descendant, we adopt a sequential approach similar to Pandey et al. (2024) to model p_{prop}. Given a descendant, we order its progenitor features $\mathcal{X}_{\text{p}}$ in descending mass orders and sequentially predict each new progenitor $\mathbf{x}_{\text{p},i}$ using the previous progenitors $\{\mathbf{x}_{\text{p},1} \ldots \mathbf{x}_{\text{p},i-1}\}$. The NDE consists of a GRU for embedding the progenitor sequence into a latent space and a conditional neural spline flow (Durkan et al., 2019) for density modeling. We denote flow density for $\mathbf{x}_{\text{p},i}$ as $\hat{q}(\mathbf{x}_{\text{p},i} \mid \{\mathbf{x}_{\text{p},0} \ldots \mathbf{x}_{\text{p},i-1}\}, \mathcal{C})$, where the additional condition $\mathcal{C}$ includes the encoded history, progenitor redshift z_{p}, and progenitor count N_{p}. Here, we also introduce a zero starting token $\mathbf{x}_{\text{p},0} = \vec{\mathbf{0}}$ to predict the first progenitor.

1.2. Training & Generation

We train all three model components simultaneously. Training is performed halo by halo: given a merger tree, at each training step, we select a random descendant halo and compute the corresponding loss (see also Appendix A.2).

Given a descendant $\mathbf{x}_{\text{d}}$ with N_{p} progenitors and features $\{\mathbf{x}_{\text{p},1} \ldots \mathbf{x}_{\text{p},N_{\text{p}}}\}$, the classifier loss is the cross-entropy loss, $\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{N_{\max}} \pi_i \log \hat{\pi}_i$, where π_i is the one-hot encod-

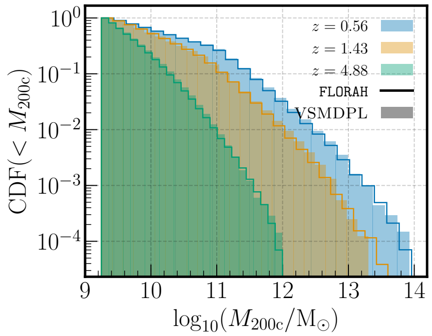

Figure 2. The normalized, cumulative mass functions of VSMDPL (shaded) and FLORAH-Tree (solid) for three redshift slices.

ing of N_p. The NDE loss is the negative log-likelihood summed over all progenitors, $\mathcal{L}_\mathrm{NDE} = -\sum_{i=1}^{N_\mathrm{p}} \log \hat{q}(\mathbf{x}_{\mathrm{p},i} \mid \{\mathbf{x}_{\mathrm{p},1} \ldots \mathbf{x}_{\mathrm{p},i-1}\}, \mathcal{C})$. The final loss function is $\mathcal{L}_\mathrm{total} = \mathcal{L}_\mathrm{NDE} + \alpha \mathcal{L}_\mathrm{CE}$, where α controls the relative weight between the losses. In practice, we find $\alpha = 1$ works well.

We use an AdamW optimizer (Loshchilov & Hutter, 2019; Kingma & Ba, 2014) with a peak learning rate 10^{-5} and weight decay coefficient 0.01. The learning rate follows a cosine annealing schedule (Loshchilov & Hutter, 2016), with $500,000$ warm-up steps and $25,000$ decay steps. We use a batch size of 128. Training converges after approximately 72 hours on a single NVIDIA Tesla A100 GPU.

Given a list of root halos and redshifts, generation proceeds redshift by redshift: we process all halos within each redshift before proceeding to the next, while tracking all descendant-progenitor connections. This enables efficient batching, with consistent input dimensionality despite the varying halo count per snapshot. For each descendant, we encode its history, then randomly sample the progenitor count from the classifier's multinomial distribution $\hat{\pi}$ and progenitor features from the NDE $\hat{q}$. After iterating over all redshifts, we reconstruct the trees using the recorded descendant-progenitor connections.

We terminate progenitor branches when their sampled mass falls below a min mass threshold of $1.7 \times 10^6\,\mathrm{M}_\odot$ (200 particles). We also enforce the descending mass ordering. If sampled progenitors violate this condition, we simply resample, though, in practice, we find such occurrences extremely rare with sufficient training data.

For testing, we take the 50,000 root halos in the test set and randomly generate a distinct merger tree realization for each halo. The full generation process takes ~ 16 minutes on a single GPU (with reconstruction accounting for ~ 6 minutes), significantly lower than N-body simulations and easily parallelizable for further efficiency gains.

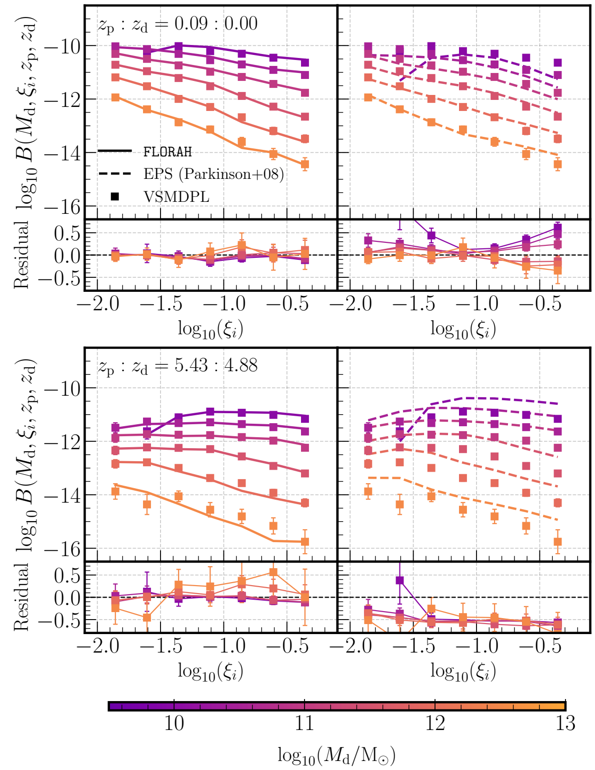

Figure 3. Merger rates as a function of progenitor mass ratios. The left panels compare FLORAH-Tree and VSMDPL merger rate, while the right panels compare EPS-based and VSMDPL merger rates. Error bars representing Poisson uncertainties of VSMDPL.

Results & Conclusion

Fig. 1 shows examples of the generated merger trees. Each column shows FLORAH-Tree and VSMDPL merger trees with the same root mass, demonstrating FLORAH-Tree captures merger tree diversity in VSMDPL. Fig. 2 shows the normalized, cumulative mass functions for three redshift slices, demonstrating good agreement between FLORAH-Tree and VSMDPL.

We now compare FLORAH-Tree against merger trees generated using the traditional Extended Press-Schechter (EPS) formalism. Specifically, we employ the implementation of Parkinson et al. (2008) as provided by SatGen (Jiang et al., 2021). For consistency, we apply identical post-processing to FLORAH-Tree and EPS trees: a minimum halo mass of $200\,M_\mathrm{dm}$ and a maximum progenitor count of $N_\mathrm{prog} = 3$.

Fig. 3 compares the merger rates of FLORAH-Tree, EPS-based, and VSMDPL trees for two redshift slices. Following Fakhouri & Ma (2008), we define the merger rate as the number of mergers per unit volume, descendant mass, and redshift. The progenitor mass ratio is the mass of each non-primary progenitor relative to the primary progenitor. The results show strong agreement between the merger rates

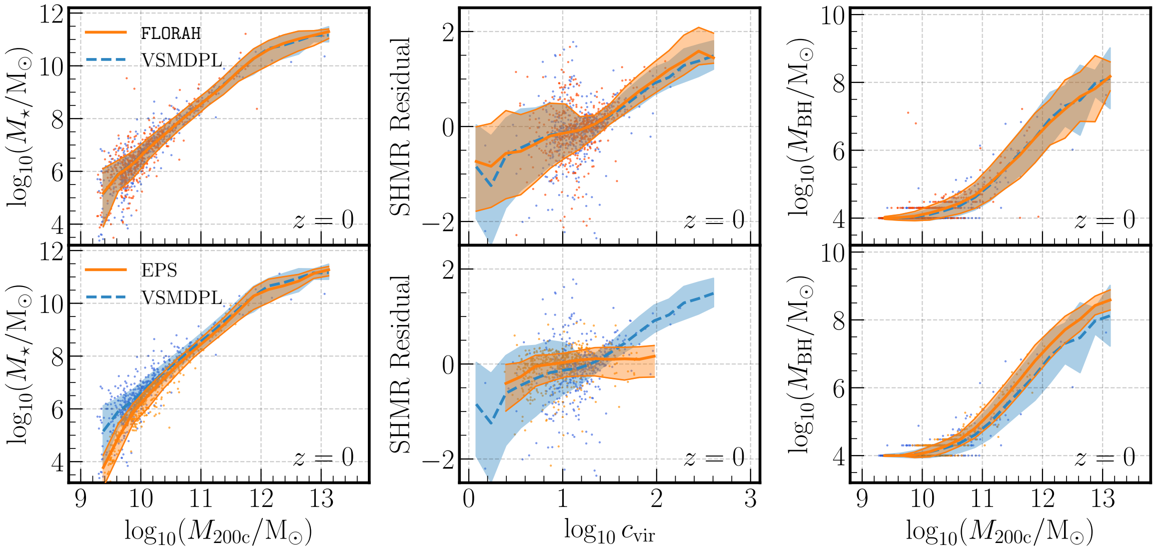

Figure 4. Comparison of scaling relations from SC-SAM predictions on `FLORAH-Tree`, EPS-based, and VSMDPL merger trees at $z = 0$. Top panels compare `FLORAH-Tree` and VSMDPL merger trees, while bottom panels compare the EPS-based and VSMDPL merger trees. From left to right, the columns show the SHMR, SHMR residual-concentration relation, and the SMBH mass-halo mass relation. Solid lines and bands represent the median and 68% intervals.

predicted by `FLORAH-Tree` and those from VSMDPL, with consistent performance across mass and redshift bins. In contrast, the EPS trees systematically overpredict merger rates, consistent with prior studies (e.g., Zhang et al., 2008).

Perhaps the most stringent test of our generated trees is the predictions they yield for galaxy properties when combined with commonly used recipes for baryonic physics. We apply the Santa Cruz SAM (SC-SAM, Somerville et al. 2008) to the generated trees and assign semi-observable galaxy properties onto each DM halo. Specifically, we compute the stellar-to-halo-mass relation (SHMR), the SHMR residual-concentration relation, and the supermassive black hole (SMBH) mass-halo mass relation. In SC-SAM, both SMBH growth (through merger-driven "bright mode" and steady "radio mode" accretion) and stellar mass assembly (through merger-driven starbursts and quiescent star formation) depend critically on the accretion and merger histories. Lastly, the SHMR residual-concentration relation probes galaxy assembly bias by showing stellar mass dependencies beyond halo mass (e.g. Gabrielpillai et al., 2022).

Fig. 4 compares the median and 68% confidence intervals of these relations for `FLORAH-Tree`, EPS-based, and VSMDPL at $z = 0$. For all scaling relations, the `FLORAH-Tree` predictions align remarkably well with the VSMDPL results. The SAMs run on EPS-based trees recover the SHMR reasonably well, albeit with noticeable discrepancies at low halo masses. However, the EPS-based trees show poor agreement with the SHMR residual-concentration relation, displaying minimal correlation between SHMR residual and concentration. This lack of correlation highlights the well-known limitation of EPS-based algorithms in capturing the dependence of merger history on secondary halo properties. Additionally, the EPS SAM run systematically overpredicts SMBH masses, consistent with its overestimation of merger rates in Fig. 3.

To conclude, `FLORAH-Tree` represents a significant step toward fast, accurate merger tree generation that can enable large-scale galaxy formation studies previously computationally prohibitive with N-body simulations alone. Most notably, the model demonstrates consistent performance across wide mass and redshift ranges. This is particularly significant given the mass imbalance in the training dataset (i.e., the shape of the halo mass function) and the potential for error accumulation seen in autoregressive techniques. The accurate reproduction of key galaxy-halo scaling relations with SC-SAM confirms that `FLORAH-Tree` generates physically realistic merger trees suitable for downstream galaxy formation modeling, while substantially outperforming traditional EPS-based approaches.

More broadly, our framework can be generalized to a broad class of hierarchical tree-generative problems in astrophysics and beyond. Future extensions include leveraging multiple simulations to expand the dynamic range of merger trees, incorporating information from the simulation's initial density fields, and conditioning on cosmological parameters using suites like CAMELS (Villaescusa-Navarro et al., 2021) and DREAMS (Rose et al., 2025).

Software and Data

The GitHub repository for `FLORAH-Tree` and instructions for downloading the pre-trained models and a subset of the generated merger trees can be found at https://github.com/trivnguyen/florah-tree.

This project uses the following software: `IPython` (Perez & Granger, 2007), `Jupyter` (Kluyver et al., 2016), `Matplotlib` (Hunter, 2007), `NumPy` (Harris et al., 2020), `PyTorch` (Paszke et al., 2019), `PyTorch Geometric` (Fey & Lenssen, 2019), `PyTorch Lightning` (Falcon et al., 2020), `SciPy` (Virtanen et al., 2020), `SatGen` (Jiang et al., 2021), `zuko` (Rozet et al., 2024), `ytree` (Smith & Lang, 2019).

Acknowledgements

We thank Viraj Pandya and Christian Kragh Jespersen for discussions on EPS-based merger trees, and Claude-André Faucher-Giguère, Tjitske Starkenburg, Matthew Ho, Justine Zeghal, Sammy Sharief, Ronan Legin, Laurence Perreault-Levasseur for helpful feedback.

TN and SM are supported by the National Science Foundation under Cooperative Agreement PHY-2019786 (The NSF AI Institute for Artificial Intelligence and Fundamental Interactions, http://iaifi.org/). TN is also supported by a CIERA Postdoctoral Fellowship. RS is supported by the Flatiron Institute. The Flatiron Institute is supported by the Simons Foundation. CM is supported by James Arthur Postdoctoral Fellowship at CCPP.

The CosmoSim database is a service provided by the Leibniz Institute for Astrophysics Potsdam (AIP). The MultiDark database was developed in cooperation with the Spanish MultiDark Consolider Project CSD2009-00064. The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) and the Partnership for Advanced Supercomputing in Europe (PRACE, www.prace-ri.eu) for funding the MultiDark simulation project by providing computing time on the GCS Supercomputer SuperMUC at Leibniz Supercomputing Centre (LRZ, www.lrz.de). The computations for this work were, in part, run at facilities supported by the Scientific Computing Core at the Flatiron Institute, a division of the Simons Foundation. The data used in this work were, in part, hosted on equipment supported by the Scientific Computing Core at the Flatiron Institute, a division of the Simons Foundation.

References

Behroozi, P. S., Wechsler, R. H., and Wu, H.-Y. The rockstar phase-space temporal halo finder and the velocity offsets of cluster cores. *The Astrophysical Journal*, 762(2):109, dec 2012a. doi: 10.1088/0004-637X/762/2/109. URL https://dx.doi.org/10.1088/0004-637X/762/2/109.

Behroozi, P. S., Wechsler, R. H., Wu, H.-Y., Busha, M. T., Klypin, A. A., and Primack, J. R. Gravitationally consistent halo catalogs and merger trees for precision cosmology. *The Astrophysical Journal*, 763(1):18, dec 2012b. doi: 10.1088/0004-637X/763/1/18. URL https://dx.doi.org/10.1088/0004-637X/763/1/18.

Bower, R. G. The evolution of groups of galaxies in the Press-Schechter formalism. *MNRAS*, 248:332–352, January 1991. doi: 10.1093/mnras/248.2.332.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv e-prints*, art. arXiv:1406.1078, June 2014. doi: 10.48550/arXiv.1406.1078.

Chuang, C.-Y., Jespersen, C. K., Lin, Y.-T., Ho, S., and Genel, S. Leaving No Branches Behind: Predicting Baryonic Properties of Galaxies from Merger Trees. *ApJ*, 965 (2):101, April 2024. doi: 10.3847/1538-4357/ad2b6c.

Correa, C. A., Wyithe, J. S. B., Schaye, J., and Duffy, A. R. The accretion history of dark matter haloes - III. A physical model for the concentration-mass relation. *MNRAS*, 452(2):1217–1232, September 2015. doi: 10.1093/mnras/stv1363.

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. Neural Spline Flows. *arXiv e-prints*, art. arXiv:1906.04032, June 2019. doi: 10.48550/arXiv.1906.04032.

Fakhouri, O. and Ma, C.-P. The nearly universal merger rate of dark matter haloes in ΛCDM cosmology. *MNRAS*, 386(2):577–592, May 2008. doi: 10.1111/j.1365-2966.2008.13075.x.

Falcon, W. et al. Pytorchlightning/pytorch-lightning: 0.7.6 release, May 2020. URL https://doi.org/10.5281/zenodo.3828935.

Fey, M. and Lenssen, J. E. Fast Graph Representation Learning with PyTorch Geometric. *arXiv e-prints*, art. arXiv:1903.02428, March 2019. doi: 10.48550/arXiv.1903.02428.

Gabrielpillai, A., Somerville, R. S., Genel, S., Rodriguez-Gomez, V., Pandya, V., Yung, L. Y. A., and Hernquist, L. Galaxy formation in the Santa Cruz semi-analytic model compared with IllustrisTNG – I. Galaxy scaling relations, dispersions, and residuals at z = 0. *Monthly Notices of the Royal Astronomical Society*, 517(4):6091–6111, 08 2022. ISSN 0035-8711. doi: 10.1093/mnras/stac2297. URL https://doi.org/10.1093/mnras/stac2297.

Hahnloser, R. H. R., Sarpeshkar, R., Mahowald, M. A., Douglas, R. J., and Seung, H. S. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, 405(6789):947–951, June 2000. doi: 10.1038/35016072.

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020.

Hendrycks, D. and Gimpel, K. Gaussian Error Linear Units (GELUs). *arXiv e-prints*, art. arXiv:1606.08415, June 2016. doi: 10.48550/arXiv.1606.08415.

Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Computing in Science and Engineering*, 9(3):90–95, May 2007. doi: 10.1109/MCSE.2007.55.

Hutter, A., Dayal, P., Yepes, G., Gottlöber, S., Legrand, L., and Ucci, G. Astraeus I: the interplay between galaxy formation and reionization. *MNRAS*, 503(3):3698–3723, May 2021. doi: 10.1093/mnras/stab602.

Jespersen, C. K., Cranmer, M., Melchior, P., Ho, S., Somerville, R. S., and Gabrielpillai, A. Mangrove: Learning Galaxy Properties from Merger Trees. *ApJ*, 941(1):7, December 2022. doi: 10.3847/1538-4357/ac9b18.

Jiang, F., Dekel, A., Freundlich, J., van den Bosch, F. C., Green, S. B., Hopkins, P. F., Benson, A., and Du, X. SatGen: a semi-analytical satellite galaxy generator - I. The model and its application to Local-Group satellite statistics. *MNRAS*, 502(1):621–641, March 2021. doi: 10.1093/mnras/staa4034.

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

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., Willing, C., and Jupyter Development Team. Jupyter Notebooks—a publishing format for reproducible computational workflows. In *IOS Press*, pp. 87–90. 2016. doi: 10.3233/978-1-61499-649-1-87.

Klypin, A., Yepes, G., Gottlöber, S., Prada, F., and Heß, S. MultiDark simulations: the story of dark matter halo concentrations and density profiles. *Monthly Notices of the Royal Astronomical Society*, 457(4):4340–4359, 02 2016. ISSN 0035-8711. doi: 10.1093/mnras/stw248. URL https://doi.org/10.1093/mnras/stw248.

Liu, C., Fan, W., Liu, Y., Li, J., Li, H., Liu, H., Tang, J., and Li, Q. Generative Diffusion Models on Graphs: Methods and Applications. *arXiv e-prints*, art. arXiv:2302.02591, February 2023. doi: 10.48550/arXiv.2302.02591.

Liu, J., Kumar, A., Ba, J., Kiros, J., and Swersky, K. Graph Normalizing Flows. *arXiv e-prints*, art. arXiv:1905.13177, May 2019. doi: 10.48550/arXiv.1905.13177.

Loshchilov, I. and Hutter, F. SGDR: Stochastic Gradient Descent with Warm Restarts. *arXiv e-prints*, art. arXiv:1608.03983, August 2016. doi: 10.48550/arXiv.1608.03983.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Navarro, J. F., Frenk, C. S., and White, S. D. M. The Structure of Cold Dark Matter Halos. *ApJ*, 462:563, May 1996. doi: 10.1086/177173.

Nguyen, T., Modi, C., Yung, L. Y. A., and Somerville, R. S. FLORAH: a generative model for halo assembly histories. *MNRAS*, 533(3):3144–3163, September 2024. doi: 10.1093/mnras/stae2001.

Pandey, S., Modi, C., Wandelt, B. D., Bartlett, D. J., Bayer, A. E., Bryan, G. L., Ho, M., Lavaux, G., Makinen, T. L., and Villaescusa-Navarro, F. CHARM: Creating Halos with Auto-Regressive Multi-stage networks. *arXiv e-prints*, art. arXiv:2409.09124, September 2024. doi: 10.48550/arXiv.2409.09124.

Parkinson, H., Cole, S., and Helly, J. Generating dark matter halo merger trees. *MNRAS*, 383(2):557–564, January 2008. doi: 10.1111/j.1365-2966.2007.12517.x.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *arXiv e-prints*, art. arXiv:1912.01703, December 2019. doi: 10.48550/arXiv.1912.01703.

Perez, F. and Granger, B. E. IPython: A System for Interactive Scientific Computing. *Computing in Science and Engineering*, 9(3):21–29, Jan 2007. doi: 10.1109/MCSE.2007.53.

Planck Collaboration. Planck 2013 results. XVI. Cosmological parameters. *A&A*, 571:A16, November 2014. doi: 10.1051/0004-6361/201321591.

Press, W. H. and Schechter, P. Formation of Galaxies and Clusters of Galaxies by Self-Similar Gravitational Condensation. *ApJ*, 187:425–438, February 1974. doi: 10.1086/152650.

Robles, S., Gómez, J. S., Ramírez Rivera, A., Padilla, N. D., and Dujovne, D. A deep learning approach to halo merger tree construction. *MNRAS*, 514(3):3692–3708, August 2022. doi: 10.1093/mnras/stac1569.

Rose, J. C., Torrey, P., Villaescusa-Navarro, F., Lisanti, M., Nguyen, T., Roy, S., Kollmann, K. E., Vogelsberger, M., Cyr-Racine, F.-Y., Medvedev, M. V., Genel, S., Anglés-Alcázar, D., Kallivayalil, N., Wang, B. Y., Costanza, B., O'Neil, S., Roche, C., Karmakar, S., Garcia, A. M., Low, R., Lin, S., Mostow, O., Cruz, A., Caputo, A., Farahi, A., Muñoz, J. B., Necib, L., Teyssier, R., Dalcanton, J. J., and Spergel, D. Introducing the DREAMS Project: DaRk mattEr and Astrophysics with Machine Learning and Simulations. *ApJ*, 982(2):68, April 2025. doi: 10.3847/1538-4357/adb8e5.

Rozet, F., Divo, F., and Schnake, S. probabilists/zuko: Zuko 1.1.0, January 2024.

Smith, B. D. and Lang, M. ytree: A python package for analyzing merger trees. *Journal of Open Source Software*, 4(44):1881, dec 2019. doi: 10.21105/joss.01881. URL https://doi.org/10.21105/joss.01881.

Somerville, R. S. and Davé, R. Physical Models of Galaxy Formation in a Cosmological Framework. *ARA&A*, 53:51–113, August 2015. doi: 10.1146/annurev-astro-082812-140951.

Somerville, R. S., Lemson, G., Kolatt, T. S., and Dekel, A. Evaluating approximations for halo merging histories. *MNRAS*, 316(3):479–490, August 2000. doi: 10.1046/j.1365-8711.2000.03467.x.

Somerville, R. S., Hopkins, P. F., Cox, T. J., Robertson, B. E., and Hernquist, L. A semi-analytic model for the co-evolution of galaxies, black holes and active galactic nuclei. *Monthly Notices of the Royal Astronomical Society*, 391(2):481–506, 11 2008. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2008.13805.x. URL https://doi.org/10.1111/j.1365-2966.2008.13805.x.

Tang, K. S. and Ting, Y.-S. Galaxy Merger Reconstruction with Equivariant Graph Normalizing Flows. In *Machine Learning for Astrophysics*, pp. 13, July 2022. doi: 10.48550/arXiv.2207.02786.

van den Bosch, F. C. The universal mass accretion history of cold dark matter haloes. *MNRAS*, 331(1):98–110, March 2002. doi: 10.1046/j.1365-8711.2002.05171.x.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need, 2023.

Villaescusa-Navarro, F., Anglés-Alcázar, D., Genel, S., Spergel, D. N., Somerville, R. S., Dave, R., Pillepich, A., Hernquist, L., Nelson, D., Torrey, P., Narayanan, D., Li, Y., Philcox, O., La Torre, V., Maria Delgado, A., Ho, S., Hassan, S., Burkhart, B., Wadekar, D., Battaglia, N., Contardo, G., and Bryan, G. L. The CAMELS Project: Cosmology and Astrophysics with Machine-learning Simulations. *ApJ*, 915(1):71, July 2021. doi: 10.3847/1538-4357/abf7ba.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1. 0 Contributors. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, February 2020. doi: 10.1038/s41592-019-0686-2.

Wechsler, R. H., Zentner, A. R., Bullock, J. S., Kravtsov, A. V., and Allgood, B. The Dependence of Halo Clustering on Halo Formation History, Concentration, and Occupation. *ApJ*, 652(1):71–84, November 2006. doi: 10.1086/507120.

White, S. D. M. and Rees, M. J. Core condensation in heavy halos: a two-stage theory for galaxy formation and clustering. *MNRAS*, 183:341–358, May 1978. doi: 10.1093/mnras/183.3.341.

You, J., Ying, R., Ren, X., Hamilton, W. L., and Leskovec, J. GraphRNN: Generating Realistic Graphs with Deep Auto-regressive Models. *arXiv e-prints*, art. arXiv:1802.08773, February 2018. doi: 10.48550/arXiv.1802.08773.

Yung, L. Y. A., Somerville, R., Finkelstein, S., Popping, G., and Davé, R. Semi-analytic forecasts for JWST - I. UV luminosity functions at z = 4-10. *MNRAS*, 483(3):2983–3006, March 2019. doi: 10.1093/mnras/sty3241.

Yung, L. Y. A., Somerville, R. S., Nguyen, T., Behroozi, P., Modi, C., and Gardner, J. P. Characterising ultra-high-redshift dark matter halo demographics and assembly histories with the GUREFT simulations. *MNRAS*, May 2024. doi: 10.1093/mnras/stae1188.

Zhang, J., Fakhouri, O., and Ma, C.-P. How to grow a healthy merger tree. *MNRAS*, 389(4):1521–1538, October 2008. doi: 10.1111/j.1365-2966.2008.13671.x.

Name	Parameters
History Encoder	4.13×10^5
Neural Density Estimator	5.70×10^5
Classifier	3.37×10^4
Total	$\mathbf{1.02 \times 10^6}$

Table 1. Breakdown of the number of trainable parameters in each model component.

A. Additional details on the machine learning framework

A.1. Forward model

As described in Section 1.1, the forward model consists of three primary components: (1) a history encoder to encode the history parameter and redshift feature $\mathcal{H}_{\rm hist}$ and $\mathcal{Z}_{\rm hist}$, (2) a classifier that takes in this encoded history and predicts a multinomial probability vector of the progenitor counts $\hat{\pi}$, and (3) an NDE to model the distribution of progenitor features $\mathcal{X}_{\rm p}$. Here we provide more details about the network architecture of each model component.

The history encoder consists of four GRU layers, each with 128 hidden units. All but the last layer followed by a ReLU activation (Hahnloser et al., 2000). During the forward pass, both $\mathcal{H}_{\rm hist}$ and $\mathcal{Z}_{\rm hist}$ are independently projected into a 128-D space using two separate FC layers. The resulting projections are summed and then passed through the GRU layers. Only the output vector from the final timestep in the sequence is used as the encoded representation.

The classifier takes in the 128-D encoded history and outputs a multinomial probability vector $\hat{\pi}$. The vector $\hat{\pi}$ has a dimension of $N_{\max}$, the max progenitor counts, which is set to $N_{\max} = 3$ in this study. The classifier consists of four FC layers, each with 16 hidden units and a GELU activation (Hendrycks & Gimpel, 2016). To account for the progenitor redshift, $z_{\rm p}$, we project $z_{\rm p}$ into a 128-D space using an FC layer and then add it to encoded history. The resulting representation is then passed through the classifier. A softmax function is applied to the output of the final layer to produce $\hat{\pi}$.

As mentioned, to model the properties of a varying number of progenitor, we order the targeted progenitor features $\mathcal{X}_{\rm p}$ in descending mass orders, i.e. $\mathcal{X}_{\rm p} = \{\mathbf{x}_{{\rm p},1} \ldots \mathbf{x}_{{\rm p},N}\}$ where $M_{{\rm p},i} > M_{{\rm p},j}$ for $i < j$. To model the distribution of $\mathbf{x}_{{\rm p},i}$, we first use a GRU encoder (referred to as the progenitor encoder) to embed the previous progenitors $\{\mathbf{x}_{{\rm p},0}, \mathbf{x}_{{\rm p},1} \ldots \mathbf{x}_{{\rm p},i-1}\}$ into a 128-D latent space. Here $\mathbf{x}_{{\rm p},0} = \tilde{\mathbf{0}}$ is a zero starting token. We then feed this encoded representation as conditions for a normalizing flow. The GRU progenitor encoder has a similar architecture to the history encoder, i.e. four GRU layers, each with 128 hidden units. For the flow, we use four neural spline flows with monotonic rational-quadratic spline transformations, each with 8 knots and a hidden size of 128.

Additionally, we condition the flow on the progenitor count $N_{\rm p}$. We represent $N_{\rm p}$ as a one-hot encoded vector and project it onto a 128-D latent space using a single FC layer. This projected representation is added to the encoded progenitor representation before being fed as conditioning input to the flow. We find this conditioning to be essential, likely due to mass conservation: $N_{\rm p}$ determines how the descendant halo's mass is partitioned among progenitors. Systems with many progenitors predominantly undergo minor mergers, while those with few progenitors involve major merger events. This relationship between progenitor multiplicity and mass distribution has been extensively studied (e.g. Somerville et al., 2000).

The full model has a total of 1.02×10^6 trainable parameters. Table 1 provides the breakdown of each model component.

Lastly, we experiment with the Transformer architecture (Vaswani et al., 2023). In this manner, the history encoder functions as the Transformer encoder, while the progenitor encoder in the NDE serves as the decoder. We also experiment with different ways to encode the redshift $\mathcal{Z}_{\rm hist}$ information such as positional sinusoidal encoding (Vaswani et al., 2023). We find that performance remains largely unchanged, with the GRU-based model being computationally faster.

A.2. Training

As mentioned in Section 1.1, the training process proceeds halo by halo: given a merger tree, in each training step, we randomly select a descendant halo and its progenitors and compute the loss function. In practice, this can be computationally inefficient. Thus, at each training step, instead of select a single halo of a merger tree, we select a random branch and compute the loss for on all descendant halos within that branch. This approach leverages the sequential nature of the GRU layers more effectively. It also naturally places more weight on halos at lower redshifts, as halos at these redshifts appear multiple times for a single merger tree. This is important for autoregressive models since, during generation, any errors at

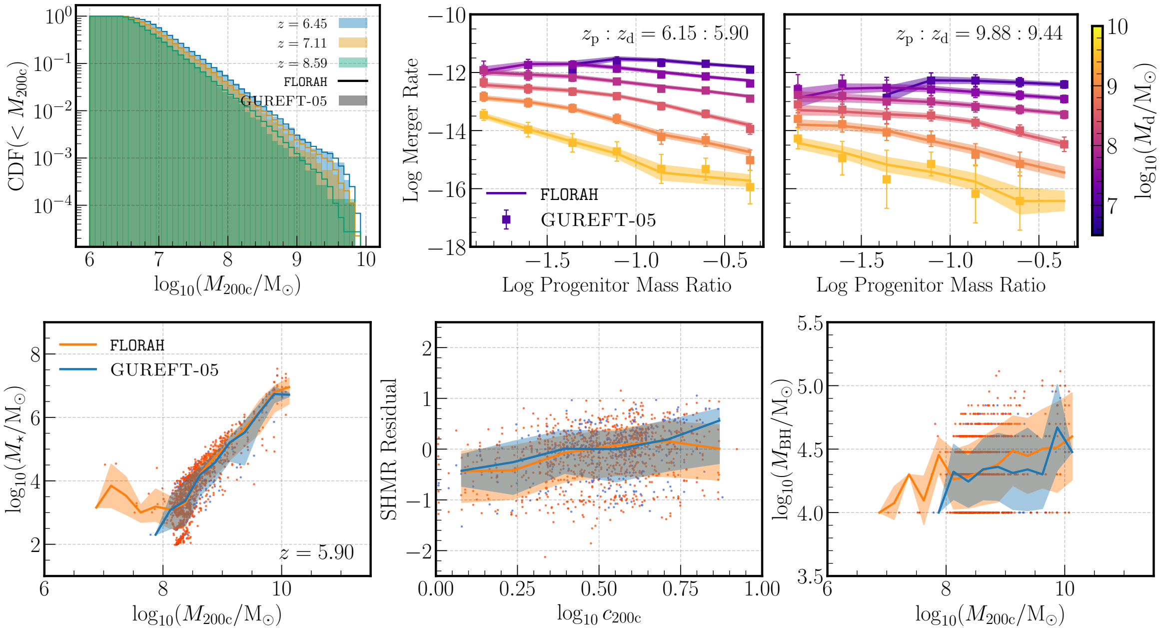

Figure 5. Top left: The normalized, cumulative mass functions of `FLORAH-Tree` (solid) and GUREFT-05 (shaded) for three redshift slices. *Top right*: Merger rates as a function of progenitor mass ratios. `FLORAH-Tree` results are shown as solid lines with bands, GUREFT-05 as error bars. Bands and error bars representing Poisson uncertainties. *Bottom*: Comparison of scaling relations from SC-SAM predictions on `FLORAH-Tree` (orange) and GUREFT-05 (blue) merger trees at $z = 5.90$. Solid lines and bands represent the median and 68% intervals.

early time steps will propagate to later time steps.

B. Additional result on the GUREFT simulation

The GUREFT (Gadget at Ultrahigh Redshift with Extra-Fine Timesteps; Yung et al. 2024) simulations are a suite of N-body simulations designed to probe the assembly histories of DM halos at ultra-high redshift ($z = 5.9 - 40$). A distinctive feature of GUREFT is its unprecedented temporal resolution, with 171 simulation snapshots between $z = 5.9 - 40$, spaced at one-tenth of the halo dynamical time. This enables detailed reconstruction of merger histories during the earliest cosmic times. We present results for the GUREFT-05 box, which contains the highest number of halos and merger trees. The simulation volume is $(5\,\mathrm{Mpc}\,h^{-1})^3$ and contains 1024^3 particles at a mass resolution of $1.5 \times 10^4 \mathrm{M}_\odot$.

Similarly, we apply a resolution cut of 500 particles ($7.5 \times 10^5 \mathrm{M}_\odot$) for the root halos at $z = 5.9$ and 200 particles ($3.0 \times 10^5 \mathrm{M}_\odot$) for progenitor halos. After the cut, we obtained approximately 35,000 merger trees, which we then split into subsets of 20,000, 8000, and 7000 trees for the training, validation, and test datasets, respectively.

Training proceeds similarly, using the same architecture described in Sections 1.1 and A.1. For inference, we take root halos of the 7,000 merger trees in the test dataset and generate 5 independent realizations for each root halo to increase statistical robustness. This results in approximately 35,000 total merger trees in the `FLORAH-Tree` dataset. We subsequently apply SC-SAM to both the generated and simulation merger trees to derive stellar masses and SMBH masses for each halo.

Fig. 5 shows the comparison between `FLORAH-Tree` and GUREFT-05 merger trees. Panels are formatted similarly to Fig. **??**: the top left panel, top right panels, and bottom panels display the normalized cumulative mass function, merger rates, and the three scaling relations, respectively. Consistent with our previous findings, `FLORAH-Tree` successfully reproduces these key statistics in agreement with GUREFT-05. The agreement is somewhat reduced compared to the VSMDPL case, likely attributable to the smaller size of the training dataset.

In case of SC-SAM results, while the test and `FLORAH-Tree` datasets contain 7,000 and 35,000 merger trees respectively, only 320 and 1,588 trees possess assigned stellar masses at $z = 5.90$. As a result, the scaling relations predicted by `FLORAH-Tree` extend marginally beyond those of GUREFT-05. This difference represents a statistical sampling effect rather than a systematic discrepancy in the underlying merger statistics.