Skepthical review: *QTT-Based Compression of Merger Tree Trajectories for Assembly Bias Studies: A Proof-of-Concept with Dummy Implementation*

Summary

The manuscript presents a proof-of-concept pipeline for applying Quantum Tensor Trains (QTT) to dark matter halo merger tree trajectories with the goal of studying assembly bias. Starting from 1000 merger trees from a cosmological simulation, the authors extract main progenitor branches, engineer node-level features (log mass, concentration, $v_{\max}$, scale factor) with per-tree normalization, pad trajectories to a fixed length, and reshape them for QTT decomposition. The intended compressed representations would then feed linear regression and MLP models to predict a $z \approx 0$ halo property, and are compared with baselines that use only final-snapshot features, which achieve $R^2 \approx 0.41$–0.44.

However, the qttpy library was unavailable, so all QTT-related steps rely on a dummy implementation that effectively outputs random, non-informative features. Consequently, reconstruction errors are ≈ 1 regardless of rank, compression ratios scale trivially with rank, and QTT-based predictive models perform poorly with large negative R^2, making all QTT results artifacts of the dummy behavior. The only physically meaningful quantitative results are the baselines using final-state features, which establish the amount of variance left unexplained and thus the potential headroom for assembly-history information.

The main current contribution is therefore methodological: the paper documents an end-to-end pipeline from merger trees to trajectories, (intended) QTT tensorization, and downstream regression, and it discusses limitations, computational considerations, and future extensions, including real QTT experiments, alternative targets, trajectory baselines beyond QTT, and use of the full 5099-tree dataset. Nonetheless, the absence of real QTT experiments, limited direct assembly-bias analysis, missing implementation details, and some structural and editorial issues significantly limit the present scientific impact and reproducibility.

Strengths

- Clearly and repeatedly acknowledges the key limitation that all QTT-related operations use a dummy implementation, avoiding overstated scientific claims about QTT performance (Abstract; Sec. 1; Sec. 4.2–4.3.2; Sec. 4.7–4.8; Conclusions).
- Defines a clean, logically organized pipeline from raw merger trees to main progenitor trajectories, feature engineering, padding/reshaping, intended QTT decomposition, and regression modeling, with reasonable preprocessing choices (log mass, per-tree normalization, explicit trajectory lengths) (Sec. 2.1–2.3; Sec. 4.1).

- Provides quantitative baselines using final-state halo features alone ($R^2 \approx 0.41$–0.44) and interprets them as indicating substantial unexplained variance and possible headroom for assembly history information (Sec. 2.4.4; Sec. 4.3.1; Sec. 4.6).
- Includes a structured discussion of limitations and future work, outlining concrete next steps such as using a real QTT library, expanding the dataset to 5099 trees, exploring additional targets and main-progenitor definitions, and comparing with other trajectory-modeling methods (Sec. 4.7–4.8).
- Uses visualization (PCA/t-SNE) and a conceptual computational-cost discussion to reason about information content and scalability, even though empirical conclusions are constrained by the dummy QTT implementation (Sec. 4.4–4.5).
- The figures collectively provide clear, well-labeled visualizations that directly support the manuscript's claims, with effective use of comparative layouts, intuitive axes, and consistent alignment between figure content and narrative (e.g., Figures 1, 3, 4, 5, 6, 7, 8, and 9).
- Core preprocessing formulas are standard and internally consistent: log-transform of mass and per-tree Z-score normalization are stated with clear symbols (x, μ, σ).
- Tensor reshaping for QTT is dimensionally consistent in the reported concrete case: (128 time steps $\times$ 4 features) reshaped to $(2, 2, 2, 2, 2, 2, 2, 4)$ preserves total element count (512).
- Padding narrative (original lengths ≤ 98; then padded to 98; then to 128 as next power of 2) is logically consistent with the later stated reshape.

Major issues

1. **The central methodological component—QTT-based compression of merger tree trajectories—is never actually evaluated on real QTT features because all QTT-related computations use a dummy qttpy implementation that returns effectively random outputs (Abstract; Introduction; Sec. 2.3; Sec. 4.2; Sec. 4.3.2; Sec. 4.4; Sec. 4.7; Conclusions).** As a result, there is no empirical evidence that QTT can compress trajectories effectively, preserve assembly-bias-relevant information, or improve predictions over final-state baselines, so the main scientific claim about QTT's promise remains entirely untested.

   Recommendation: Either (i) reframe and rewrite the manuscript explicitly as a pipeline/baseline or technical note, making clear in the Title, Abstract, Introduction, and throughout Secs. 2–4 that no claims are made about QTT performance on real data and that all QTT numbers are artifacts of a dummy implementation; or (ii) postpone submission until a functional QTT implementation (e.g., qttpy or another TT/QTT package) is available and re-run the full analysis to report real reconstruction errors, compression ratios, and predictive results. In the latter case, substantially revise Secs. 4.2–4.4 and 4.6–4.8 to present genuine QTT experiments and conclusions.

2. **The connection between the implemented pipeline and specific assembly-bias science questions is weak and largely qualitative (Introduction; Sec. 2.1.2; Sec. 4.3.1; Sec. 4.6).** The exact physical identity of the target variable (first component of the $\boldsymbol{y}$ vector) is not clearly stated, nor is it justified as a probe of assembly bias. Beyond noting that baseline R^2 leaves ~ 56–59% variance unexplained, there is no quantitative analysis linking residuals to assembly history (e.g., formation times, concentration histories) or standard assembly-bias diagnostics at fixed mass.

   Recommendation: In Sec. 2.1.2, clearly define the target halo property (e.g., concentration, $v_{\rm max}$, or another quantity), explain why it is expected to be assembly-bias-sensitive, and justify focusing on the first component of $\boldsymbol{y}$ instead of alternatives. In Sec. 4.6, add at least a minimal assembly-bias analysis independent of QTT: for example, compute correlations between simple trajectory summaries (formation redshift, half-mass time, time-averaged concentration) and the target at fixed mass, and/or compare early- vs late-forming halo subsets. Relate these diagnostics and the mass-conditioned R^2 to the unexplained variance, and explicitly state that, with dummy QTT, no firm assembly-bias inference from QTT features is yet possible.

3. **The experimental comparison is limited to final-snapshot baselines and dummy-QTT-derived features with linear regression and a small MLP (Sec. 2.4; Sec. 4.3).** There is no evaluation of alternative trajectory-compression or sequence-modeling approaches (e.g., PCA on time series, autoencoders, RNNs/temporal CNNs), making it impossible to assess whether QTT—once functional—would be competitive or offer distinct advantages over standard methods.

   Recommendation: Extend Sec. 2.4 and Sec. 4.3 to include at least one or two non-QTT trajectory baselines that can be implemented immediately: for example, (i) apply PCA or an autoencoder to the padded trajectory matrices and regress on the resulting low-dimensional codes, and/or (ii) train a simple recurrent or 1D convolutional neural network that ingests full trajectories. Report these models' performance alongside the current baselines in Sec. 4.3. This will contextualize future QTT-based gains and already illuminate how much assembly-history information standard methods can exploit.

4. **Key methodological aspects of the QTT pipeline are under-specified, which will hinder reproducibility once a real QTT library is used.** In particular, the mapping from the (length $\times$ 4) trajectory to the QTT tensor shape, the two-stage padding from variable length to 98 and then to 128, the treatment of the feature dimension (4) in the tensorization, and how the compressed representation is constructed from the QTT object (cores vs full reconstruction) are not fully and consistently described (Sec. 2.3.1–2.3.2; Sec. 4.1–4.2).

   Recommendation: In Sec. 2.3 and Sec. 4.2, give a precise tensorization scheme: explicitly state that original trajectories of length 60–98 are first padded with zeros to length 98 and then to 128 ($= 2^7$) for QTT; clarify whether the four features form an

additional mode (yielding, e.g., shape $(2, 2, 2, 2, 2, 2, 2, 4)$) and whether padding is applied only along the temporal dimension. Describe step by step how the feature vector used in Sec. 2.4.1 is obtained from the QTT object (e.g., using full() followed by flattening, or concatenating core parameters). Ensure this description is internally consistent across Secs. 2.3.1–2.3.2 and 4.1–4.2.

5. **The main progenitor extraction and predictive-model setup lack sufficient implementation detail for independent reproduction (Sec. 2.2–2.4; Sec. 4.3.1–4.3.2).** For progenitors, handling of multiple branches, mask_main ambiguities, and any discarded trees are not fully specified. For the ML models, the precise MLP architecture, activation functions, loss, optimizer settings, training epochs, regularization, random seeds, data split strategy, and any preprocessing of the target are missing or only briefly mentioned.

   Recommendation: Augment Sec. 2.2 with a clear algorithmic description of main progenitor identification (e.g., how mask_main and edge_index are used, how ties between multiple progenitors are resolved, and whether any trees were dropped due to inconsistencies). In Sec. 2.4.2–2.4.3 and Sec. 4.3, document the full ML setup: number of hidden layers and units, activation functions, output layer, loss function, optimizer type and learning rate, batch size, number of epochs, early stopping criteria, regularization (dropout, weight decay), and data split strategy including random seed and whether splits are stratified. Where feasible, report variability (e.g., mean ± std R^2 over several random splits) in Sec. 4.3 to quantify stability.

6. **The section structure is inconsistent, with a stray "# 3. RESULTS" heading followed immediately by "# 4. RESULTS AND INTERPRETATION" that contains the actual results subsections, and at least one apparently incomplete or malformed subsection (e.g., a lone "####" line and a truncated sentence in Sec. 2.2–2.4) (Sec. 2.2; Sec. 2.4.4; Sec. 3; Sec. 4).** This disrupts the logical flow and could confuse readers and indexers about where the Results really start.

   Recommendation: Rationalize the section structure so that there is a single coherent Results section with consistent numbering, for example by renaming "4. RESULTS AND INTERPRETATION" to "3 Results and Interpretation" and renumbering its subsections as 3.1–3.9, or by removing the empty "3. RESULTS" heading if Section 4 is to remain as the main results. Fix malformed headings (e.g., remove stray "####" markers) and ensure that truncated text in Sec. 2.4.4 and elsewhere is restored so that all methods and results paragraphs are complete and readable.

7. **Several figures (notably Figures 1, 3, 4, 5, 6, 7, 8, and 9) have inconsistencies or omissions in captions and methodological details, such as contradictory descriptions of preprocessing steps (e.g., padding in Figure 1), lack of explicit binning parameters (Figure 1), unclear or inconsistent metric definitions and aggregation (Figure 3), missing sample sizes, and insufficient**

reporting of preprocessing or hyperparameters (e.g., t-SNE in Figure 9). Additionally, some figures lack uncertainty quantification (e.g., Figures 5, 6, 7), making it difficult to assess statistical reliability.

Recommendation: Revise captions and figure content to ensure consistency with the methods section (e.g., clarify padding workflow in Figure 1), explicitly report all relevant parameters (binning, error metrics, sample sizes, preprocessing, and hyperparameters), and add uncertainty quantification (error bars, confidence intervals, or distributional plots) where appropriate. For Figure 9, include all t-SNE settings and demonstrate embedding stability across seeds/hyperparameters.

8. **Some figures use suboptimal or misleading visual encodings, such as inappropriate use of box plots for deterministic or nearly identical distributions (Figures 3 and 4), unclear axis scales or formatting (Figures 3, 4, 5, 6), and layouts that compress important regions or obscure differences (e.g., dynamic range issues in Figures 5 and 6). Accessibility and readability are also compromised by small font sizes, low resolution, and non-colorblind-safe palettes in several figures.**

   Recommendation: Adopt more appropriate visual encodings (e.g., bar charts or overlaid distributions for deterministic data), clarify axis scales and formatting, and adjust layouts to highlight key differences (e.g., insets or broken axes for compressed regions). Increase figure resolution and font sizes, use colorblind-safe palettes, and ensure all figures are accessible and legible in print and on screen.

Minor issues

1. The title and some phrasing in the Abstract, Introduction, and Conclusions emphasize "QTT-Based Compression" and assembly bias, which can be read as implying a substantive evaluation of QTT-based compression, even though only baseline models and dummy-QTT failures are quantitatively analyzed (Title; Abstract; Introduction; Sec. 4.3.1; Conclusions).

   Recommendation: Adjust the Title and key summary statements to make the scope unambiguous, for example by explicitly mentioning "pipeline" and/or "dummy implementation" (e.g., "Pipeline for QTT-Based Compression… with Dummy Implementation"). In the Abstract and Conclusions, state early that QTT is only implemented via a placeholder and that no conclusions about real QTT performance are drawn, while foregrounding the baseline and pipeline contributions.

2. The description of the dataset and merger trees is relatively generic and omits important contextual details such as the specific simulation used, cosmological parameters, mass resolution, and selection criteria for the 1000-tree subset out of 5099 available trees (Sec. 2.1; Sec. 4.1).

Recommendation: Expand Sec. 2.1 / Sec. 4.1 to specify: (i) which N-body or hydrodynamical simulation provides the merger trees, including basic cosmology and mass resolution; (ii) how the 1000 trees were selected from the 5099 available (e.g., mass range, redshift range, random subsample); and (iii) any quality cuts applied, such as minimum number of snapshots or progenitors. State explicitly whether all selected trees yielded valid trajectories or whether some were discarded.

3. The explanation of main progenitor trajectory extraction is high-level and does not fully describe how branching, multiple progenitors, missing links, or ambiguous mask_main indices are handled, which could affect the resulting trajectories and potential biases (Sec. 2.2.1; Sec. 4.1).

   Recommendation: In Sec. 2.2.1, provide a more precise description of the main progenitor algorithm: e.g., specify whether at each step you follow the most massive progenitor, use mask_main indices directly, or apply another tie-breaking rule; clarify how missing or ambiguous indices are handled; and state whether any halos were removed due to inconsistencies. If all 1000 trees produced valid trajectories (as suggested in Sec. 4.1), explicitly mention that and why.

4. Citation formatting is inconsistent and sometimes malformed, with repeated years such as "*(Ye and Loureiro, 2024, 2024)*" and "*(Ye and Loureiro, 2024b, 2024b)*", and mixed citation styles in the References (e.g., bracketed entries like "[Elahi, P. J., …]") (Introduction; Sec. 2.3.2; References).

   Recommendation: Standardize citation formatting across the manuscript: remove duplicated years so each in-text reference appears as, for example, "(Ye and Loureiro 2024)", and only use suffixes (2024a/2024b) if genuinely distinct works exist and are clearly listed. In the References, adopt a single consistent style (e.g., unbracketed author-year) and ensure one-to-one correspondence between in-text citations and reference-list entries.

5. The description of trajectory padding and reshaping for QTT in Sec. 2.3.1–2.3.2 is somewhat ambiguous and not fully aligned with later text in Sec. 4.1–4.2 mentioning padding first to length 98 and then to 128 and giving a shape like $(\mathbf{2,2,2,2,2,2,2,4})$. It is unclear exactly how many padding steps occur and how the feature dimension is incorporated.

   Recommendation: Clarify Sec. 2.3.1–2.3.2 to match Sec. 4.1–4.2 by explicitly stating: (i) the original trajectory length range; (ii) that trajectories are first padded to a uniform length of 98 and then further padded to 128 ($\mathbf{2^7}$) for QTT; and (iii) how the four features are included in the tensor shape (e.g., as a separate mode to obtain $(\mathbf{2,2,2,2,2,2,2,4})$). Ensure the text is self-consistent and reflects the actual implementation.

6. Section 2.4.4 (Baseline Comparison) appears truncated in the provided text (e.g., line break around "Wang et al., 2023"), and the description of how baselines are trained/evaluated is slightly unclear (Sec. 2.4.4).

   Recommendation: Review Sec. 2.4.4 in the source manuscript to ensure that all sentences are complete and that the citation to Wang et al. (2023) is correctly formatted on a single line. If any details of the baseline training/evaluation procedure were inadvertently omitted, reinsert them so that the section reads as a coherent paragraph.

7. Some interpretive statements about unexplained variance being "potentially attributable to assembly bias" do not distinguish assembly history from other sources of scatter, since no conditioning on mass or other covariates is performed (Abstract; Sec. 4.3.1; Sec. 4.6; Conclusions).

   Recommendation: Qualify these statements throughout by explicitly acknowledging that the unexplained variance reflects a combination of assembly history, environmental effects, stochasticity, and modeling noise. Where feasible, add a mass-conditioned or mass-binned R^2 analysis in Sec. 4.3.1 to better isolate potential assembly-bias-related variance, and then refer to those results when discussing what might be attributable to assembly bias.

8. The latent-space visualization section currently describes mainly what is "expected" from PCA and t-SNE but provides limited explicit description of what is actually observed in the plots (e.g., gradients, clustering, or lack thereof for dummy QTT features) (Sec. 4.4).

   Recommendation: Revise Sec. 4.4 to report concrete qualitative or simple quantitative observations from the PCA and t-SNE plots: for example, note whether baseline features show a visible gradient of the target along a principal component, whether t-SNE reveals clustering, and confirm that dummy-QTT projections appear structureless with respect to the target. Optionally, include simple summary statistics such as correlations between principal components and the target.

9. The computational-cost discussion is conceptual and includes specific numbers such as "$< \mathbf{0.001}$ ms" and "4–5 ms per tree" without stating whether these are measured from the dummy implementation or theoretical estimates, and without specifying hardware or software context (Sec. 4.5).

   Recommendation: In Sec. 4.5, explicitly distinguish measured timings from rough estimates, and, for measured numbers, specify the hardware (CPU/GPU model, RAM) and software environment. If some values are extrapolations for a real QTT implementation, label them as such and present them as order-of-magnitude expectations rather than precise benchmarks.

10. Many figures omit secondary but important details, such as sample sizes (e.g., Figure 1, Figure 7, Figure 9), summary statistics overlays (Figure 1), explicit whisker conventions (Figures 3, 4), target normalization or units (Figures 5, 6, 7, 8, 9), and evalua-

tion protocols (Figures 5, 6, 7). Some figures lack clear legends, concise labeling, or panel identification, and several do not specify preprocessing steps or random seeds for reproducibility.

Recommendation: Add sample sizes, summary statistics, and whisker conventions to captions or plots; annotate target normalization/units and evaluation protocols; provide clear legends and panel labels; and specify preprocessing steps and random seeds for all relevant figures.

11. Visual clarity and accessibility are sometimes reduced by small or inconsistent typography, cramped layouts, overplotting, non-uniform axis limits, and insufficient annotation of colorbars or colormaps. Some figures do not harmonize style or formatting with the rest of the manuscript.

    Recommendation: Increase font and marker sizes, adjust plot margins and spacing, use transparency or density overlays to address overplotting, enforce consistent axis limits and tick formatting, clearly label colorbars with variable names and units, and harmonize visual style across all figures.

12. The paper interprets a "relative reconstruction error ≈ 1.0" as meaning the reconstruction is as different as a zero trajectory would be, but no reconstruction-error definition is provided, so this implication cannot be verified from the paper alone.

    Recommendation: Explicitly define the reconstruction error used (e.g., $|\boldsymbol{X} - \hat{\boldsymbol{X}}|/|\boldsymbol{X}|$ with specified norm and whether computed per-trajectory, per-feature, including/excluding padded zeros), then re-check the interpretation that error ≈ 1 corresponds to a zero reconstruction or uncorrelated output.

13. "Compression ratio" is discussed and compared across ranks, but no analytic definition is given (parameter count formula / storage cost for QTT cores vs original tensor). This prevents internal verification of claims like "ratios equal the rank" being an artifact.

    Recommendation: Define compression ratio precisely (e.g., original element count divided by number of stored parameters in TT/QTT cores) and specify how core sizes and ranks are counted, including the handling of the final feature dimension.

Very minor issues

1. There are several minor formatting and typographical problems, including mismatched quotation marks around code/library names (e.g., "\texttt{qttpy}"), stray punctuation such as a period before a citation ("library. (Ye and Loureiro, 2024b, 2024b)."), HTML-escaped symbols like "< 0.001 ms", inconsistent notation for R^2 (e.g., "R^2" vs "$\mathbb{R}^2$"), and spacing inconsistencies around inline math and commas (e.g., "(mass, concentration, $V_{\max}$, scale factor)") (Abstract; Sec. 1; Sec. 2.3.2; Sec. 4.3.1; Sec. 4.5; Conclusions).

Recommendation: Perform a thorough proofreading of the manuscript and LaTeX source to standardize formatting: use consistent quotation or \texttt{} for code-like names (e.g., \texttt{qttpy}, \texttt{mask_main}, \texttt{edge_index}); remove stray periods before citations; adopt a single notation for R^2 (e.g., "R^2") throughout text and figure captions; replace HTML escapes ("<") with proper LaTeX symbols ("$<$"); and tidy spacing around inline math, commas, and parentheses.

2. Some headings and captions exhibit minor style inconsistencies, such as a lone "####" line before "#### 2.2.1 Main Progenitor Identification", differing capitalization between "# 3. RESULTS" and "# 4. RESULTS AND INTERPRETATION", and captions that begin inline with surrounding text or are separated from their figures (Sec. 2.2; Sec. 3; Sec. 4.1–4.2; Sec. 4.3.2).

   Recommendation: Clean the LaTeX/markup so that all section and subsection headings use consistent levels and capitalization according to the target style (e.g., \section, \subsection, \subsubsection). Remove stray heading markers (such as a lone "####"), ensure that each figure caption immediately follows its figure and is on its own line, and consider slightly rephrasing captions (e.g., for Table 1 in Sec. 4.3.2) to clearly indicate that they report performance with dummy QTT features.

3. Terminology and capitalization for some physical quantities and acronyms are not fully consistent, for example "$V_{\max}$" vs "$v_{\max}$", "Log Mass" vs "log mass", and repeated re-expansion of acronyms like "Quantum Tensor Trains (QTT)" after they have already been introduced (Sec. 2.1.1; Sec. 4.1–4.3).

   Recommendation: Standardize terminology throughout: choose a consistent style for quantities such as "$V_{\max}$" and "log mass" and use it everywhere; introduce acronyms like QTT once (e.g., in the Introduction) and then use the acronym alone thereafter unless style guidelines require otherwise; and ensure capitalization is uniform in headings and body text.

4. Minor wording, style, and formatting inconsistencies are present, such as ambiguous or imprecise caption phrasing (e.g., Figure 1), mixed numerical precision, inconsistent capitalization and notation (e.g., 'vs.'/'vs', 'R2'/'R^2'), and redundant or unclear panel references.

   Recommendation: Standardize caption language, numerical precision, capitalization, and notation across all figures; clarify panel references and streamline titles and legends.

5. Some figures could further improve visual polish by adjusting gridline prominence, exporting as vector graphics, optimizing bar/marker ordering, and providing accessibility annotations or alt text.

   Recommendation: Lighten or remove unnecessary gridlines, export figures as high-resolution or vector graphics, order bars/markers logically, and add accessibility notes or alt text summarizing key takeaways.

6. Explanation for non-zero means in aggregated normalized features is potentially incomplete/ambiguous. If normalization statistics (μ, σ) are computed over all nodes in the full tree but aggregation is over main-progenitor nodes only, non-zero means are expected; if statistics are computed over the same aggregated set, the global mean should be (approximately) zero.

   Recommendation: Clarify precisely which node set is used to compute μ and σ (all nodes in the tree vs only main progenitor nodes vs padded trajectories) and which node set is used in the aggregated histograms.

7. Variable naming/notation is inconsistent (e.g., "masstransformed", "xnormalized" appear as code-style identifiers rather than mathematical symbols; "R2" vs "R^2"), which can obscure definitions.

   Recommendation: Introduce consistent mathematical notation (e.g., $m' = \log_{10}(m + \epsilon)$, $x' = (x - \mu)/\sigma$, R^2) and map code identifiers to symbols once.

8. The log-transform uses $\log_{10}(\text{mass} + 10^{-6})$ without clarifying the units/scale of 'mass'; taking a logarithm of a dimensionful quantity is formally ill-defined unless mass is implicitly in fixed units or normalized.

   Recommendation: State the mass unit/normalization convention (e.g., mass expressed in a fixed unit so the argument of $\log_{10}$ is effectively dimensionless, or mass divided by a reference mass).

9. The stated baseline R^2 summary range "≈ 0.41–0.44" is only consistent with the underlying values after rounding to 2 decimal places (LR 0.4064 rounds to 0.41; MLP 0.4351 rounds to 0.44), while strict (unrounded) inclusion would exclude 0.4064.

   Recommendation: Clarify the rounding convention explicitly when presenting the range (e.g., "$R^2 \approx 0.41$–0.44 (rounded to 2 d.p.)") or report the unrounded min/max directly.

Key statements and references

- ⚠ **The dataset used in this study consists of 1000 merger trees obtained from a cosmological simulation, specifically constructed high-fidelity halo merger trees such as those described for AbacusSummit-like simulations, which encode the hierarchical growth of dark matter halos over cosmic time and provide node features, edge indices, edge attributes, target variables, and metadata.**
- *Reference(s):* Tweed et al., 2009, Bose et al., 2022
- *Justification:* Bose et al., 2022 describes constructing high-fidelity halo merger trees from AbacusSummit and tracking progenitors/descendants over time, and Tweed et al., 2009 explains merger trees that encode hierarchical halo growth from N-body simulations. However, neither paper states that a dataset of 1000 merger trees was used,

nor do they describe a dataset formatted with node features, edge indices, edge attributes, target variables, and metadata. Thus, only the general idea of high-fidelity merger trees encoding hierarchical growth is supported.

- **✖ Each node's physical properties—mass, concentration, maximum circular velocity ($v_{\max}$), and scale factor—are engineered following established merger-tree feature choices, with mass undergoing a $\log_{10}$ transformation to handle its wide dynamic range and all features subsequently normalized per tree using Z-score normalization ($x_{\text{normalized}} = (x - \mu)/\sigma$) to stabilize variance and improve downstream analyses.**
- *Reference(s):* Parkinson et al., 2007, Robles et al., 2019, Rouhiainen et al., 2021
- *Justification:* None of the attached papers describe engineering node features as mass, concentration, $v_{\max}$, and scale factor with $\log_{10}$ mass transformation and per-tree Z-score normalization. Parkinson et al., 2007 discuss halo masses and plot conditional mass functions versus $\log_{10}(M_1/M_2)$, but not feature engineering or normalization. Robles et al., 2019 represent trees with mass, distance to main branch, and progenitor type (no concentration, $v_{\max}$, or scale factor) and do not describe $\log_{10}$ mass preprocessing or Z-score normalization. Rouhiainen et al., 2021 concerns normalizing flows for cosmological fields, not merger-tree feature choices or normalization.

- **✖ The target variable used for supervised learning is a halo property at $z = 0$ extracted from each merger tree in a manner consistent with prior work on the impact of halo merger trees on galaxy properties and on generative models for halo assembly histories, where the raw target has shape (1,2) and only the first component is retained and cast to a float tensor.**
- *Reference(s):* Lee et al., 2014, Gómez et al., 2021, Nguyen et al., 2024
- *Justification:* Neither Lee et al., 2014 nor Gómez et al., 2021 describe a supervised learning setup, a target variable, or any tensor/datatype details. While both papers analyze halo/galaxy properties (often at $z = 0$) derived from merger trees, they do not specify targets for ML, mention generative ML models for halo assembly histories, or anything like a raw target of shape $(1, 2)$ with a retained first component cast to a float tensor.

- **✖ Quantum Tensor Train (QTT) decomposition of padded merger-tree trajectories is performed using the qttpy library's ALS-based qtt.als routine, with trajectories reshaped into tensors of shape $(2, 2, 2, ...)$ under the assumption that the length N is a power of two, and with different target ranks (2, 4, and 8) explored to control the trade-off between compression and accuracy as motivated by prior QTT applications to high-dimensional kinetic equations.**
- *Reference(s):* Ye and Loureiro, 2024, Ye and Loureiro, 2024b

- *Justification:* Neither Ye and Loureiro, 2024 nor Ye and Loureiro, 2024b mention merger-tree trajectories, the qttpy library, or an ALS-based qtt.als routine. Ye and Loureiro, 2024 discusses QTT/QTC for Vlasov–Maxwell PDEs, using binary reshaping with $N = d^L$ (often $d = 2$) and the quimb library, and algorithms such as density-matrix compression, TDVP, and DMRG—not ALS. They also explore bond dimensions like 16–128, not specifically ranks 2, 4, or 8. Thus the stated workflow is not supported by the attached papers.

Mathematical consistency audit

This section audits **symbolic/analytic** mathematical consistency (algebra, derivations, dimensional/unit checks, definition consistency).

Maths relevance: light

The manuscript contains limited explicit mathematics: a $\log_{10}$ mass transform, Z-score normalization, padding/reshaping logic for QTT input, and qualitative interpretations of reconstruction error, compression ratio, and R^2. No detailed QTT/ALS derivations are included, and key metric definitions (reconstruction error, compression ratio) are omitted, limiting strict internal verification of some interpretive statements.

Checked items

1. ✔ **Log-mass transform formula** (Sec. 2.1.1, p.2 ("masstransformed = $\log_{10}(\mathrm{mass} + 10^{-6})$"))
    - **Claim:** Mass is transformed via $\log_{10}(\mathrm{mass} + 10^{-6})$ to stabilize variance and avoid $\log(0)$.
    - **Checks:** algebra/symbol correctness, domain/sanity, units/dimensional consistency
    - **Verdict:** PASS; confidence: medium; impact: minor
    - **Assumptions/inputs:** mass is nonnegative (or at least $\mathrm{mass} + 10^{-6} > 0$)., 10^{-6} is in the same units as mass (or mass is unitless).
    - **Notes:** Algebra and intent are consistent. Formal unit issue: log of a dimensionful quantity is not well-defined unless mass is in fixed units or normalized; paper does not specify this (flagged as very minor clarity issue).
2. ✔ **Per-tree Z-score normalization definition** (Sec. 2.1.1, p.2 ("xnormalized = $(x - \mu)/\sigma$"))
    - **Claim:** Each feature is normalized within each tree by subtracting its mean and dividing by its standard deviation.
    - **Checks:** definition consistency, units/dimensional consistency, edge-case sanity
    - **Verdict:** PASS; confidence: high; impact: minor

- **Assumptions/inputs:** $\sigma \neq 0$ for each feature within each tree (or special handling if $\sigma = 0$)., μ and σ are computed over a specified node set (not fully specified later).
- **Notes:** Standard Z-score normalization; yields dimensionless variables. The paper does not discuss $\sigma = 0$ edge cases.

3. ⚠ **Non-zero mean after aggregation explanation** (Sec. 4.1, p.4 (discussion of aggregated normalized feature distributions))

   - **Claim:** Aggregated normalized features have non-zero means due to per-tree normalization followed by aggregation.
   - **Checks:** logic consistency, definition consistency
   - **Verdict:** UNCERTAIN; confidence: medium; impact: minor
   - **Assumptions/inputs:** Normalization is performed per tree using some node set; aggregation is over nodes from main progenitor branches across trees.
   - **Notes:** If μ and σ are computed on exactly the same nodes later aggregated (per tree), then each tree's normalized node-mean is ~ 0, implying the global aggregated mean should also be ~ 0 (up to numerical error). Non-zero aggregated means are consistent if (i) μ, σ computed over all nodes in the tree but aggregation is only main-progenitor nodes, or (ii) padded zeros are included post-normalization, or (iii) a different weighting/selection is used. The paper does not specify which node set is used for μ, σ versus aggregation.

4. ✔ **Padding to maximum trajectory length then power-of-2** (Sec. 2.3.1, p.3 and Sec. 4.1, p.4)

   - **Claim:** Trajectories are padded with zeros to a uniform length (max observed), then further padded to the next power of two for QTT.
   - **Checks:** logic consistency, shape/dimension consistency
   - **Verdict:** PASS; confidence: high; impact: minor
   - **Assumptions/inputs:** Padding uses zeros appended in time dimension., The QTT implementation benefits from or requires power-of-2 sizes.
   - **Notes:** Narrative is consistent: observed max length reported as 98 (p.4), then padded to 98, then to 128 ($= 2^7$).

5. ✔ **QTT reshape element-count consistency** (Sec. 4.2, p.4 ("(128 time steps x 4 features) reshaped into (2,2,2,2,2,2,2,4)"))

   - **Claim:** The padded trajectory tensor is reshaped for QTT to $(2, 2, 2, 2, 2, 2, 2, 4)$.
   - **Checks:** shape/dimension consistency, algebraic check (products)
   - **Verdict:** PASS; confidence: high; impact: moderate
   - **Assumptions/inputs:** Original stored tensor has 128×4 entries per trajectory.

- **Notes:** $128 \times 4 = 512$ entries; $2^7 \times 4 = 512$ entries. The reshape is consistent.

6. ⚠ **General QTT reshape description (N is power of 2)** (Sec. 2.3.2, p.3 (trajectory reshaped to $(n_1, ..., n_d)$ with product $= N$; assume N is power of $2 \rightarrow (2, 2, ...)$))

   - **Claim:** Trajectory length N is assumed to be a power of 2 and reshaped into a $(2, 2, 2, ...)$ tensor.
   - **Checks:** notation/definition consistency, logic consistency
   - **Verdict:** UNCERTAIN; confidence: medium; impact: minor
   - **Assumptions/inputs:** N refers to the number of entries being quantized (unclear if time steps only or time $\times$ features).
   - **Notes:** Later, the paper reshapes a (128×4) object into $(2^7, 4)$, effectively treating the quantized dimension as 128 and leaving features as 4. The earlier statement "product equals trajectory length N" is ambiguous for multivariate trajectories: trajectory length could mean time steps (128) or total entries (512). Clarify what N denotes.

7. ⚠ **Reconstruction error interpretation at value ~ 1** (Sec. 4.2, p.4-5 (discussion around Figure 3))

   - **Claim:** A relative reconstruction error of ~ 1.0 implies the reconstructed trajectory is as different from the original as a zero trajectory would be, or is essentially uncorrelated noise if the original was normalized.
   - **Checks:** definition dependency check, sanity/limiting-case reasoning
   - **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
   - **Assumptions/inputs:** Reconstruction error is a relative norm like $|X - \hat{X}|/|X|$ in some norm., Normalization makes typical magnitudes comparable.
   - **Notes:** Without an explicit formula for "relative reconstruction error," the implication error $\approx 1 \Leftrightarrow$ similar to zero reconstruction cannot be validated. For example, if error $= |X - \hat{X}|/|X|$, then $\hat{X} = 0$ gives error $= 1$ exactly; but other definitions (MSE-based, featurewise averages, inclusion of padded zeros, etc.) change the interpretation.

8. ⚠ **Compression ratio qualitative claim** (Sec. 4.2, p.5 (discussion around Figure 4))

   - **Claim:** Compression ratios equal to 2, 4, 8 (matching ranks) indicate simplified parameter counting (e.g., original_size/rank).
   - **Checks:** definition dependency check, logic consistency
   - **Verdict:** UNCERTAIN; confidence: medium; impact: minor
   - **Assumptions/inputs:** Compression ratio defined as original storage divided by compressed storage.

- **Notes:** No definition of compression ratio is provided, so it is not possible to audit whether the stated values correspond to any consistent parameter-counting scheme.

9. ✔ **Baseline feature vector definition** (Sec. 4.3.1, p.5-6)
    - **Claim:** Baseline input is the last snapshot feature vector (Log Mass, Concentration, $V_{\max}$, Scale Factor).
    - **Checks:** definition consistency, shape consistency
    - **Verdict:** PASS; confidence: high; impact: minor
    - **Assumptions/inputs:** The last snapshot refers to the final scale factor along the extracted main progenitor trajectory.
    - **Notes:** Consistent with earlier definition of trajectory ordered by scale factor and selecting last point.
10. ✔ **Use of flattened full() output as QTT feature vector** (Sec. 2.4.1, p.3 and Sec. 4.3.2, p.6)
    - **Claim:** QTT representation is converted to a fixed-size vector by calling full() and flattening the result.
    - **Checks:** shape consistency, logic consistency
    - **Verdict:** PASS; confidence: medium; impact: minor
    - **Assumptions/inputs:** full() returns a reconstructed tensor of consistent shape across samples (after padding).
    - **Notes:** Given uniform padding to 128×4, flattening yields a consistent 512-dimensional vector per tree.

Limitations

- Audit is restricted to the content present in the provided PDF text; the paper contains no explicit QTT/ALS derivations or formal definitions for key metrics (reconstruction error, compression ratio), preventing full symbolic verification of those parts.
- Figures are referenced but the underlying metric formulas used to generate them are not specified in the text, so only shape/logical consistency can be checked for those discussions.

Numerical results audit

This section audits **numerical/empirical** consistency: reported metrics, experimental design, baseline comparisons, statistical evidence, leakage risks, and reproducibility.

All 22 automated numeric/logical consistency checks passed. Most checks confirm internal arithmetic/logic (splits, powers of two, tensor reshape size, percent/complement conversions, sign checks, and time-rate consistency). Some reported statistics and model metrics remain unrecomputable from text alone and are therefore only sanity-checked or treated as unverified for reproducibility.

Checked items

1. ✔ **C1_dataset_subset_fraction** (Page 4, Sec. 4.1)
    - **Claim:** "The initial dataset comprised 1000 merger trees (a subset of the full 5099 trees available)."
    - **Checks:** percentage_of_total
    - **Verdict:** PASS
    - **Notes:** Computed subset fraction $= 1000/5099 = 0.1961168857$ (19.61168857%).
2. ✔ **C2_mass_transform_constant** (Page 2, Sec. 2.1.1)
    - **Claim:** "masstransformed $= \log_{10}(\text{mass} + 10^{-6})$, where a small constant (10^{-6}) was added…"
    - **Checks:** constant_parsing
    - **Verdict:** PASS
    - **Notes:** Parsed unicode-minus exponent form as 1×10^{-6}.
3. ✔ **C3_train_val_split_counts_1000** (Page 3, Sec. 2.4.2; Page 5, Sec. 4.3)
    - **Claim:** "Data was split into training and validation sets (80% training, 20% validation)." applied to 1000 trajectories.
    - **Checks:** parts_vs_total
    - **Verdict:** PASS
    - **Notes:** Computed counts: train $= 800$, val $= 200$; sums to 1000 with integer counts.
4. ✔ **C4_trajectory_padding_to_power_of_two** (Page 4, Sec. 4.1 (also Fig. 1 caption))
    - **Claim:** "...further padded to a length of 128 (the next power of 2, 2^7)."
    - **Checks:** power_of_two_identity
    - **Verdict:** PASS
    - **Notes:** Verified $2^7 = 128$.
5. ✔ **C5_next_power_of_two_from_max_length** (Page 4, Sec. 4.1)
    - **Claim:** "original lengths up to 98… further padded to a length of 128 (the next power of 2)"
    - **Checks:** next_power_of_two
    - **Verdict:** PASS
    - **Notes:** Computed next_pow2(98)$= 128$.
6. ✔ **C6_trajectory_tensor_shape_product** (Page 4, Sec. 4.2)
    - **Claim:** "reshaped into tensors of shape $(2, 2, 2, 2, 2, 2, 2, 4)$" for 128 time steps × 4 features.

- **Checks:** shape_product_consistency
- **Verdict:** PASS
- **Notes:** Product of reshape dims $= 2^7 \times 4 = 512$, matching $128 \times 4 = 512$.

7. ✔ **C7_length_stats_ordering_and_range** (Page 4, Sec. 4.1)
   - **Claim:** "lengths range from a minimum of 60 to a maximum of 98 snapshots, with a mean trajectory length of approximately 88.5 and a median of 88.0."
   - **Checks:** stat_bounds_and_order
   - **Verdict:** PASS
   - **Notes:** Mean and median are within $[60, 98]$, and median is between min and max.

8. ✔ **C8_uniform_length_98_vs_max_98** (Page 4, Sec. 4.1)
   - **Claim:** "original lengths up to 98… padded with zeros to a uniform length of 98."
   - **Checks:** consistency_equalities
   - **Verdict:** PASS
   - **Notes:** Uniform padded length equals stated maximum original length (98).

9. ✔ **C9_aggregated_norm_mean_sd_logmass** (Page 4, Sec. 4.1)
   - **Claim:** "normalized log mass distribution showed a mean of approximately 2.07 and a standard deviation of 1.10."
   - **Checks:** nonnegativity_and_basic_sanity
   - **Verdict:** PASS
   - **Notes:** Sanity check: SD is positive ($1.10 > 0$).

10. ✔ **C10_aggregated_norm_mean_sd_conc** (Page 4, Sec. 4.1)
    - **Claim:** "normalized concentration distribution had a mean of approximately -0.21 and a standard deviation of 0.61."
    - **Checks:** nonnegativity_and_basic_sanity
    - **Verdict:** PASS
    - **Notes:** Sanity check: SD is positive ($0.61 > 0$).

11. ✔ **C11_aggregated_norm_mean_sd_vmax** (Page 4, Sec. 4.1)
    - **Claim:** "normalized $V_{\max}$ distribution exhibited a mean of about 2.09 and a standard deviation of 0.88."
    - **Checks:** nonnegativity_and_basic_sanity
    - **Verdict:** PASS
    - **Notes:** Sanity check: SD is positive ($0.88 > 0$).

12. ✔ **C12_aggregated_norm_mean_sd_scalefactor** (Page 4, Sec. 4.1)

- **Claim:** "normalized scale factor distribution had a mean of roughly 0.93 and a standard deviation of 1.45."
- **Checks:** nonnegativity_and_basic_sanity
- **Verdict:** PASS
- **Notes:** Sanity check: SD is positive ($1.45 > 0$).

13. ✔ **C13_recon_error_same_across_ranks** (Page 4, Sec. 4.2)
    - **Claim:** "mean reconstruction error… consistently approximately 1.0 across all ranks (Rank 2: ~ 1.0000000040, Rank 4: ~ 1.0000000040, Rank 8: ~ 1.0000000040)."
    - **Checks:** repeated_constant_match
    - **Verdict:** PASS
    - **Notes:** Values match across ranks (spread=0) and deviate from 1.0 by $4.000000108916879 \times 10^{-9}$.
14. ✔ **C14_compression_ratio_equals_rank** (Page 5, Sec. 4.2)
    - **Claim:** "average compression ratios… were 2.0 for rank 2, 4.0 for rank 4, and 8.0 for rank 8."
    - **Checks:** functional_relationship
    - **Verdict:** PASS
    - **Notes:** Verified compression ratio equals rank for 2, 4, and 8 as stated.
15. ✔ **C15_all_trajectories_processed_count** (Page 5, Sec. 4.2)
    - **Claim:** "All 1000 trajectories were 'successfully' processed… for all ranks, with no reported failures."
    - **Checks:** count_consistency
    - **Verdict:** PASS
    - **Notes:** Logical consistency check: failures implied 0, so successful implied 1000.
16. ✔ **C16_baseline_R2_range_from_two_models** (Page 6, Sec. 4.3.1; Page 1 abstract ($R^2 \approx 0.41 - 0.44$); Page 9 summary)
    - **Claim:** Baseline R^2 values are 0.4064 (LR) and 0.4351 (MLP), summarized as "$R^2 \approx 0.41$–0.44".
    - **Checks:** range_contains_values
    - **Verdict:** PASS
    - **Notes:** Unrounded inclusion fails for LR ($0.4064 < 0.41$), but rounding to 2 d.p. yields 0.41 and 0.44, consistent with the stated range.
17. ✔ **C17_baseline_MSE_improvement** (Page 6, Sec. 4.3.1)
    - **Claim:** Baseline MSE: LR 0.00699 vs MLP 0.00666 ("modest improvement").

- **Checks:** difference_and_percent_change
- **Verdict:** PASS
- **Notes:** Verified $0.00666 < 0.00699$; absolute improvement $= 0.00033$ and relative improvement $\approx 4.7210\%$.

18. ✔ **C18_R2_to_variance_explained_percent** (Page 6, Sec. 4.3.1)
    - **Claim:** "An R^2 of $\sim$ 0.41-0.44 means that roughly 41-44% of the variance… can be explained…"
    - **Checks:** percent_conversion
    - **Verdict:** PASS
    - **Notes:** Converted 0.41–0.44 to 41%–44% exactly.
19. ✔ **C19_unexplained_variance_from_R2_range** (Page 8, Sec. 4.6)
    - **Claim:** "This implies that 56-59% of the variance… remains unexplained…" given baseline $R^2 \approx 0.41$–0.44.
    - **Checks:** complement_to_one
    - **Verdict:** PASS
    - **Notes:** Computed unexplained: $(1 - 0.44) \times 100 = 56.0$ and $(1 - 0.41) \times 100 = 59.0$ (floating-point representation yields tiny $\sim 1 \times 10^{-14}$ residual).
20. ✔ **C20_table1_negative_R2_all_rows** (Page 6, Table 1)
    - **Claim:** Table 1 reports negative R^2 scores for all dummy-QTT models.
    - **Checks:** sign_check
    - **Verdict:** PASS
    - **Notes:** All listed dummy-QTT R^2 values are strictly negative; maximum is -1.397517.
21. ✔ **C21_dummy_vs_baseline_MSE_ratio_examples** (Page 6, Table 1 and Sec. 4.3.1)
    - **Claim:** Dummy-QTT MSE values are "significantly higher than the baseline."
    - **Checks:** ratio_comparison
    - **Verdict:** PASS
    - **Notes:** Using best baseline MSE $= 0.00666$, all dummy MSEs exceed baseline; fold-increases range from $\sim 4.2416\times$ to $\sim 112.6179\times$.
22. ✔ **C22_total_time_for_dummy_qtt** (Page 7, Sec. 4.5)
    - **Claim:** "Dummy QTT… approximately 4-5 ms per tree, totaling around 5 seconds for 1000 trees for a single rank."
    - **Checks:** time_rate_consistency
    - **Verdict:** PASS

- **Notes:** Computed total time range: $1000 \times (0.004\text{–}0.005)\text{s} = 4\text{–}5\text{s}$; claimed $\sim 5\text{s}$ is consistent.

Limitations

- Only parsed text from the provided PDF pages was used; no external data or code execution context is available.
- Values shown only in figures/plots cannot be extracted reliably without pixel-level reading; checks avoid plot-derived numbers unless explicitly stated in text.
- Most performance metrics (MSE/R^2) and distribution statistics cannot be recomputed without the underlying dataset and training/evaluation code; only algebraic/logical consistency checks are feasible.
- Some statements (e.g., distribution means/SDs and model metric values) can only be assessed for basic sanity or internal consistency, not independently reproduced from the provided inputs.