

Cosmo3DFlow: Wavelet Flow Matching for Spatial-to-Spectral Compression in Reconstructing the Early Universe

Md. Khairul Islam
mi3se@virginia.edu
Department of Computer Science
University of Virginia
Charlottesville, VA, USA

Zeyu Xia
zeyu.xia@virginia.edu
Department of Computer Science
University of Virginia
Charlottesville, VA, USA

Ryan Goudjil
kqs9vd@virginia.edu
Department of Computer Science
University of Virginia
Charlottesville, VA, USA

Jialu Wang
walter.wang@utexas.edu
Department of Statistics and Data
Sciences
The University of Texas at Austin
Austin, TX, USA

Arya Farahi
arya.farahi@austin.utexas.edu
Department of Statistics and Data
Sciences
The University of Texas at Austin
Austin, TX, USA

Judy Fox*
ckw9mp@virginia.edu
School of Data Science
Department of Computer Science
University of Virginia
Charlottesville, VA, USA

Abstract

Reconstructing the early universe from the evolved present-day universe is a challenging and computationally demanding problem in modern astrophysics. We devise a novel generative framework, Cosmo3DFlow, designed to address dimensionality and sparsity, the critical bottlenecks inherent in current state-of-the-art methods for cosmological inference. By integrating 3D Discrete Wavelet Transform (DWT) with flow matching, we effectively represent high-dimensional cosmological structures. The Wavelet Transform addresses the “void problem” by translating spatial emptiness into spectral sparsity. It decouples high-frequency details from low-frequency structures, and wavelet-space velocity fields facilitate stable ordinary differential equation (ODE) solvers with large step sizes. Using large-scale cosmological N -body simulations at 128^3 resolution, we achieve up to $46\times$ faster sampling than diffusion models. Our results enable initial conditions to be sampled in seconds, compared to minutes for previous methods.

CCS Concepts

• **Applied computing** → **Astronomy**; • **Computing methodologies** → **Neural networks**; *Learning latent representations*; *Artificial intelligence*; • **Mathematics of computing** → *Ordinary differential equations*.

Keywords

Generative models, Wavelet representation, Spatial sparsity, Spectral sparsity, Flow Matching, Astronomy

ACM Reference Format:

Md. Khairul Islam, Zeyu Xia, Ryan Goudjil, Jialu Wang, Arya Farahi, and Judy Fox. 2026. Cosmo3DFlow: Wavelet Flow Matching for Spatial-to-Spectral Compression in Reconstructing the Early Universe. In *Proceedings of the*

*Corresponding author.



This work is licensed under a Creative Commons Attribution 4.0 International License. *KDD 2026, Jeju Island, Republic of Korea.*

© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2259-2/2026/08
<https://doi.org/10.1145/3770855.3818994>

32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD 2026), August 9–13, 2026, Jeju Island, Republic of Korea. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3770855.3818994>

Resource Availability:

The source code of this paper has been made publicly available at <https://doi.org/10.5281/zenodo.20261896> and on GitHub at <https://github.com/UVA-MLSys/Cosmo3DFlow>.

1 Introduction

Understanding the origins of the universe relies on mapping the complex, non-linear structures observed today back to their initial conditions [8]. This is a *high-dimensional inverse problem* of staggering complexity, often described as trying to un-mix a fluid to determine its initial state. In the context of cosmology, the “fluid” is the distribution of dark matter and baryonic matter that forms galaxies, stars, and gas. The “mixing” is driven by gravitational instability as the universe evolves.

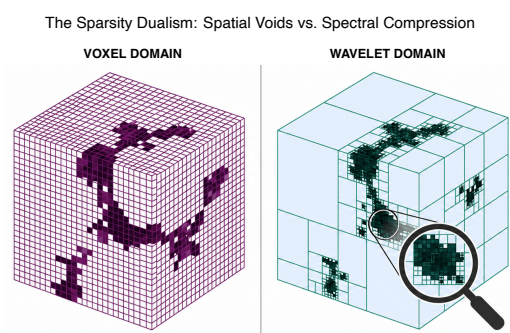


Figure 1: Cosmic Web data representations. Left: Voxel grids allocate memory uniformly, wasting computation on empty voids. Right: Wavelets concentrate informative coefficients on filaments and halos, leaving void regions as near-zero coefficients.

Analysis of N -body simulations indicates a stark disconnect between the volume a structure occupies and the mass it contains.

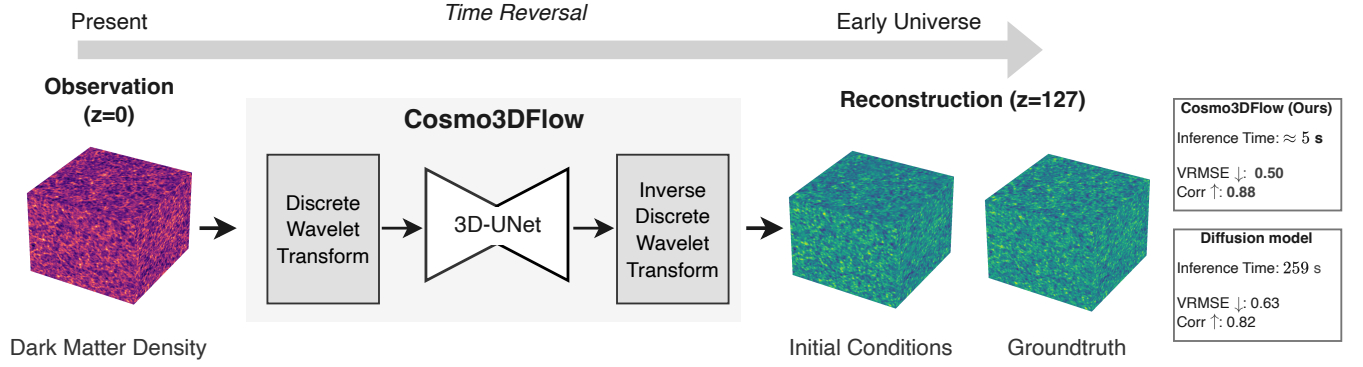


Figure 2: Cosmo3DFlow accelerates high-dimensional inverse problems with $46\times$ faster sampling. It reconstructs samples in ≈ 5 s (vs. 259 s for diffusion), preserving fine-scale structure through wavelet sparsity while achieving superior variance-normalized root mean squared error (VRMSE) and correlation metrics.

Voids are the dominant topological feature by volume: data derived from dark matter simulations indicate that voids occupy approximately 63.7% of the total volume [16]. However, despite this volumetric dominance, they effectively represent “empty” computational space, containing only 16.2% of the dark matter mass. In contrast, filaments and halos—the regions of primary astrophysical interest where galaxies form—are sparse but mass-dense. Filaments occupy roughly 4.3% of the volume but house 33.0% of the mass, while halos are even more extreme, representing a tiny volumetric fraction yet containing significant mass concentrations.

The cosmology inference challenge is formidable. For a simulation on a 128^3 grid (approximately 2.1 million voxels), a 63.7% void fraction implies that over 1.3 million voxels are dedicated to describing regions where the density contrast is low and the gradient is near zero. A typical full N -body simulation (forward model) mapping initial conditions to observations requires numerically simulating gravitational dynamics via N -body methods, where traditional Markov Chain Monte Carlo (MCMC) approaches require thousands of forward simulations per sample. In standard Convolutional Neural Networks (CNNs) or diffusion models operating in voxel space, every one of these 1.3 million void voxels requires the same number of floating-point operations (FLOPs) as the dense, complex voxels inside a galaxy halo. This is the “uniform computation cost” mismatch of the voxel grid: resources are allocated based on volume, not information density. The voxel representation forces a “worst-case scenario” resolution across the entire domain as the solver must be fine enough everywhere to handle the densest regions, leading to massive redundancy in the sparse regions (see the left panel of Figure 1). This mismatch between the uniform sampling of the Eulerian grid and the heterogeneous clustering of matter is the fundamental physical driver for adopting multi-scale representations such as hierarchical ball trees [34], 3D point clouds [2], and wavelets.

Recent advances have opened directions for high-dimensional problems. Score-based diffusion has been applied to this problem [12] but remains computationally expensive. Two recent developments in generative modeling suggest potential improvements:

(1) *Flow matching* formulates generation as solving a deterministic ordinary differential equation (ODE) rather than a stochastic process, typically enabling faster and more stable sampling with fewer integration steps [13]. (2) *Wavelet-based generative models* have demonstrated that operating in the wavelet coefficient domain rather than pixel space can provide substantial computational savings while preserving the high-frequency details crucial for many applications [4, 15]. We hypothesize that combining these two innovations is particularly well-suited for cosmological inference due to the hierarchical nature of structure formation [33].

We introduce **Cosmo3DFlow**, a framework combining flow matching with wavelet-space modeling for high-dimensional inverse problems (Figure 2). Cosmo3DFlow achieves order of magnitude speedups over diffusion baselines while matching or improving reconstruction quality. **Our main contributions include:**

- We combine the **3D Discrete Wavelet Transform** with **flow matching** for cosmological inverse problems. The DWT losslessly reshapes the field into eight wavelet sub-bands at half spatial resolution, enabling $2\times$ larger training batches and lower per-step inference cost.
- We introduce wavelet-aware scale-specific conditioning that exploits the multi-resolution structure of DWT coefficients, and power spectrum regularization that balances reconstruction error across wavenumbers during training.
- On **Quijote** 128^3 simulations, Cosmo3DFlow achieves $\approx 46\times$ faster sampling than diffusion models while matching or improving reconstruction fidelity, measured by lower variance-normalized root mean squared error (VRMSE), higher correlation, and accurate power spectrum recovery.

The methodological contribution of Cosmo3DFlow lies in the exact mathematical coupling of Optimal Transport (Flow Matching) with discrete spectral bijections. By replacing lossy neural autoencoders with the Discrete Wavelet Transform (DWT), it achieves a lossless spatial-to-spectral compression—an exact reshape under which sparse spatial structure (voids) collapses into spectrally sparse coefficients. This methodology structurally smooths the learned optimal transport vector field—bounding its Lipschitz constant to enable ultra-fast, numerically stable ODE integration. Ultimately,

Cosmo3DFlow establishes a highly efficient methodology for scaling sparse high-dimensional generative AI models for the physical sciences.

2 Related Work

The Cosmological Inference Challenge. The standard model of cosmology (Λ CDM) posits that the cosmic web evolved from initial Gaussian Random Field fluctuations [3]. Under gravitational instability, peaks of this Gaussian field collapse into virialized halos while underdense regions expand to form voids [1]. Reconstructing the initial density field, denoted as $\mathbf{x}$, from the evolved observation $\mathbf{y}$ can be framed as a Bayesian inference task. We sample from the posterior distribution $p(\mathbf{x}|\mathbf{y}) = p(\mathbf{y}|\mathbf{x})p(\mathbf{x})/p(\mathbf{y})$. Here, the likelihood $p(\mathbf{y}|\mathbf{x})$ encapsulates the forward model of gravitational evolution (typically an N -body simulation), and the prior $p(\mathbf{x})$ is the Gaussian prior of the initial conditions.

The challenge is twofold: **Dimensionality.** As resolutions increase, the dimensionality explodes. **Computational Cost.** Traditional statistical approaches like Bayesian Origin Reconstruction from Galaxies (BORG) [9, 10] utilize Hamiltonian Monte Carlo to explore this high-dimensional landscape. However, this involves running the expensive forward simulation and can take thousands of CPU hours.

Scientific significance. Mapping present-day non-linear structures back to their initial conditions is a fundamental inverse problem for probing early-universe physics (e.g., inflation, neutrino masses). Discrepancies in modeling probes like galaxy clustering and the cosmic microwave background can reveal new physics. High-fidelity suites like Quijote [32] provide the controlled environment needed to validate these methods.

The Generative AI Paradigm Shift. Recent advances in deep generative modeling offer a transformative alternative: training a neural network to approximate the posterior directly. Modi et al. [19] predicted point estimates of initial conditions via recurrent inference machines coupled with a differentiable N -body simulator; Shallue and Eisenstein [26] explored convolutional neural networks for deterministic reconstruction, and Riveros et al. [23] applied diffusion to generate 3D density fields of modified $f(R)$ gravity. Legin et al. [12] applied score-based diffusion [6, 28] to cosmological initial conditions inference, though stochastic differential equation (SDE) sampling in high-dimensional pixel space incurs substantial cost. Latent diffusion models have also been used for spatiotemporal physics emulations [24] and simulation super-resolution [17].

Flow Matching. Flow matching learns velocity fields via simple regression, offering more stable training and deterministic ODE sampling with fewer integration steps than stochastic SDEs [13]. CosmoFlow [11] demonstrated flow matching for cosmological representation learning but operated at lower resolution (2D, 256×256) without sampling full 3D volumes. FlowLensing [25] applied flow matching to 2D gravitational lensing but operated in pixel space without addressing 3D volumetric inverse problems.

Wavelet-Based Generative Models. Recent works [4, 22, 27] demonstrated that diffusion models in wavelet space enable efficient high-resolution generation. Phung et al. [22] showed that wavelet diffusion models can match or exceed the quality of latent diffusion models while avoiding the need to train auxiliary

autoencoders. Friedrich et al. [4] adapted discrete wavelet diffusion for high-resolution 3D medical imaging. Mebratu and Wu [15] used Wavelet Flow to model the field-level probability distributions of extragalactic foregrounds in cosmic observations. Valogiannis et al. [30] applied Wavelet Scattering Transform to constrain the nature of gravity using 3D N -body simulations.

3 Methodology

3.1 Problem Formulation

We formalize our goal as a conditional generation problem: given an observation $\mathbf{y} \in \mathbb{R}^{N^3}$ (evolved state), sample from the posterior $p(\mathbf{x}|\mathbf{y})$ over latent states $\mathbf{x} \in \mathbb{R}^{N^3}$ (initial conditions). The forward process $\mathbf{y} = f(\mathbf{x})$ is nonlinear and expensive to evaluate (requiring numerical simulation), making traditional MCMC intractable. Our goal is a generative model that produces high-quality samples conditioned on $\mathbf{y}$.

A natural response to this sparsity is to employ a sparse generative model that omits empty regions, as is common in natural image generation. Such an approach is ill-suited here; cosmological voids are not truly empty: they exert weak gravitational influence on surrounding structure and encode information about large-scale modes, and discarding them would remove a physically meaningful signal. We therefore retain a dense formulation, but cast it in the wavelet domain, where voids collapse into a small number of low-magnitude coefficients and are compactly represented rather than excised.

3.2 Flow Matching Framework

Flow matching learns continuous normalizing flows by regressing velocity fields that transport samples from a simple prior p_0 to the target data distribution p_1 [13]. Given samples from the prior $\mathbf{x}_0 \sim p_0$ and target $\mathbf{x}_1 \sim p_1$, we define an interpolation path:

$$\mathbf{x}_t = \mu_t(\mathbf{x}_0, \mathbf{x}_1) = t\mathbf{x}_1 + (1-t)\mathbf{x}_0, \quad t \in [0, 1] \quad (1)$$

This linear interpolation, known as rectified flow [14], yields a constant target velocity field:

$$\mathbf{u}_t = \frac{d\mathbf{x}_t}{dt} = \mathbf{x}_1 - \mathbf{x}_0 \quad (2)$$

For conditional generation with observation $\mathbf{y}$, we train a velocity network $\mathbf{v}_\theta(\mathbf{x}_t, t, \mathbf{y})$ to approximate this target velocity by minimizing the flow matching objective:

$$\mathcal{L}_{\text{flow}} = \mathbb{E}_{t \sim \mathcal{U}(0,1), \mathbf{x}_0 \sim p_0, \mathbf{x}_1 \sim p_1} [\|\mathbf{v}_\theta(\mathbf{x}_t, t, \mathbf{y}) - \mathbf{u}_t\|_2^2] \quad (3)$$

At inference time, we sample initial noise $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and integrate the learned velocity field to obtain the final sample:

$$\mathbf{x}_1 = \mathbf{x}_0 + \int_0^1 \mathbf{v}_\theta(\mathbf{x}_t, t, \mathbf{y}) dt \quad (4)$$

where the integral accumulates the velocity contributions along the flow trajectory, transporting the initial noise $\mathbf{x}_0$ to the target distribution. We approximate this integral using a numerical ODE solver with K discrete steps. By a standard Euler stability argument, the maximum stable step size scales inversely with the Lipschitz constant $\text{Lip}(\mathbf{v}_\theta)$ of the learned velocity field, so a smaller $\text{Lip}(\mathbf{v}_\theta)$

admits fewer integration steps (**Proposition 3.1**). The deterministic nature of ODE integration enables more stable and efficient generation.

3.3 Wavelet Flow Matching

Wavelet Domain Transformation. The Discrete Wavelet Transform (DWT) consists of separable convolutions with low-pass (L) and high-pass (H) filters followed by downsampling by a factor of 2 along each spatial dimension. For a 3D volume $\mathbf{y} \in \mathbb{R}^{N^3}$, this decomposition yields eight wavelet coefficient tensors:

$$\text{DWT}(\mathbf{y}) = (\mathbf{c}_{LLL}, \mathbf{c}_{LLH}, \mathbf{c}_{LHL}, \mathbf{c}_{LHH}, \mathbf{c}_{HLL}, \mathbf{c}_{HLH}, \mathbf{c}_{HHL}, \mathbf{c}_{HHH}) \quad (5)$$

where each coefficient tensor $\mathbf{c}_*$ has dimensions $(N/2)^3$. We use the Haar wavelet basis, which applies box-averaging for the low-pass filter and pairwise differencing for the high-pass filter along each axis. To verify this choice, we tested smoother alternatives (Daubechies-2, Symlet-2, Biorthogonal-2.2) from the PyWavelets library on a 32^3 grid. Haar remains optimal: it achieves the fastest per-epoch training (5.32 s vs. 5.54 s for the next best), the lowest VRMSE (0.283 vs. 0.297), and the best physical scores (e.g., cross-correlation 0.975 vs. 0.970). Across wavelet types we do not observe substantial differences in accuracy, supporting Haar as a fast and accurate default.

With all coefficients retained and consistent periodic boundary handling, this transform is exactly invertible (**Proposition 3.2**). The approximation coefficient $\mathbf{c}_{LLL}$ captures low-frequency content representing the large-scale skeleton of the cosmic web (filaments traced at coarse resolution and the interiors of voids), while the remaining seven detail coefficients encode high-frequency information aligned with halos and fine filamentary structure along different directional orientations. We stack these coefficients along the channel dimension to form a unified representation:

$$\tilde{\mathbf{y}} = \text{Stack}(\mathbf{c}_{LLL}/\sqrt{8}, \mathbf{c}_{LLH}, \dots, \mathbf{c}_{HHH}) \in \mathbb{R}^{8 \times (N/2)^3} \quad (6)$$

where we normalize $\mathbf{c}_{LLL}$ by $\sqrt{8}$ to match the dynamic range across coefficients. The original field can be exactly recovered via the Inverse Discrete Wavelet Transform (IDWT): $\mathbf{y} = \text{IDWT}(\tilde{\mathbf{y}})$, where, by convention throughout this paper, IDWT first multiplies the $\mathbf{c}_{LLL}$ channel by $\sqrt{8}$ to undo this range-matching normalization before applying the standard inverse Haar transform.

The wavelet representation offers three key advantages for cosmological inference:

- (1) **Sparse representations:** Wavelet coefficients of cosmological density fields exhibit sparse, structured distributions that are more amenable to generative modeling than dense voxel representations.
- (2) **Per-layer spatial-axis reduction:** Halving each spatial axis lets 3D convolutions after the input layer operate on volumes that are $8\times$ smaller while preserving complete information (Proposition 3.2), reducing memory and computational cost.
- (3) **Physical alignment:** The multi-resolution decomposition mirrors the hierarchical nature of cosmological structure formation, where large-scale density fluctuations are established early during inflation and small-scale structures develop through subsequent gravitational collapse.

Wavelet-Space Flow Matching. We formulate the flow matching objective entirely in wavelet space. Let $\tilde{\mathbf{x}}_0 = \text{DWT}(\mathbf{x}_0)$ and $\tilde{\mathbf{x}}_1 = \text{DWT}(\mathbf{x}_1)$ denote the wavelet-transformed noise and target fields, respectively. The wavelet-space loss becomes:

$$\tilde{\mathcal{L}}_{\text{flow}} = \mathbb{E}_{t, \tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1} [\|\mathbf{v}_\theta(\tilde{\mathbf{x}}_t, t, \tilde{\mathbf{y}}) - (\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_0)\|_2^2] \quad (7)$$

where $\tilde{\mathbf{x}}_t = t\tilde{\mathbf{x}}_1 + (1-t)\tilde{\mathbf{x}}_0$ is the interpolated state in wavelet space, and $\tilde{\mathbf{y}} = \text{DWT}(\mathbf{y})$ is the wavelet-transformed conditioning observation.

At inference time, we sample initial Gaussian noise $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ in voxel space, transform to the wavelet domain, integrate the velocity field, and apply the inverse wavelet transform to recover the final density field:

$$\hat{\mathbf{x}} = \text{IDWT} \left(\tilde{\mathbf{x}}_0 + \int_0^1 \mathbf{v}_\theta(\tilde{\mathbf{x}}_t, t, \tilde{\mathbf{y}}) dt \right) \quad (8)$$

In practice, we approximate this integral with a numerical ODE solver (e.g., Euler) over K discrete steps (Algorithm 2).

Power Spectrum Regularization. To balance reconstruction error across wavenumbers, we incorporate a log-spectral consistency loss on the velocity field. The 3D isotropic power spectrum $P(k)$ characterizes the variance of a field as a function of wavenumber $k = |\mathbf{k}|$, which is computed as:

$$P(k_i) = \frac{1}{N_{\text{modes}}(k_i)} \sum_{|\mathbf{k}| \in \text{bin}_i} |\tilde{f}(\mathbf{k})|^2 \quad (9)$$

where $\tilde{f}(\mathbf{k})$ denotes the Fourier transform of the field and $N_{\text{modes}}(k_i)$ counts modes in the i -th radial bin.

The power spectrum loss penalizes discrepancies in log-space to treat relative errors uniformly across the dynamic range:

$$\mathcal{L}_{\text{PS}} = \|\log P_{\text{pred}}(k) - \log P_{\text{target}}(k)\|_2^2 \quad (10)$$

This regularization is applied to the predicted and target velocities after they are transformed back to voxel space via the inverse DWT. The combined training objective becomes:

$$\mathcal{L}_{\text{total}} = \tilde{\mathcal{L}}_{\text{flow}} + \lambda_{\text{PS}} \mathcal{L}_{\text{PS}} \quad (11)$$

where λ_{PS} is a hyperparameter controlling the strength of the power spectrum regularization. We keep the value at 0.01.

3.4 Theoretical Analysis

We present two theoretical foundations that underpin our work: the existence of a stable lower bound on the ODE step size, and the exactness of the spatial-to-spectral transform.

PROPOSITION 3.1 (ODE STABLE STEP-SIZE LOWER BOUND). *Let the learned velocity field $\mathbf{v}_\theta(\cdot, t)$ be Lipschitz in its state argument with constant $\text{Lip}(\mathbf{v}_\theta)$. Then the explicit Euler scheme $\mathbf{z}_{k+1} = \mathbf{z}_k + \Delta t \mathbf{v}_\theta(\mathbf{z}_k, t_k)$ admits a stable step size satisfying $\Delta t \leq c/\text{Lip}(\mathbf{v}_\theta)$, where $c > 0$ depends only on the chosen stability region [5]. Consequently, the maximum stable step size scales as*

$$\Delta t_{\text{max}} \propto \frac{1}{\text{Lip}(\mathbf{v}_\theta)}, \quad (12)$$

so a velocity field with a smaller Lipschitz constant admits a larger guaranteed stable step and therefore a smaller discrete step count $K = \lceil 1/\Delta t \rceil$ for accurate trajectory integration.

PROPOSITION 3.2 (EXACTNESS OF ORTHOGONAL DWT). Let $\mathbf{x} \in \mathbb{R}^N$ be a discrete, finite-dimensional spatial input, and let $\mathbf{W} \in \mathbb{R}^{N \times N}$ denote the orthogonal discrete wavelet transform matrix. Because $\mathbf{W}^T \mathbf{W} = \mathbf{I}_N$, the forward mapping $\mathbf{z} = \mathbf{W}\mathbf{x}$ and its inverse $\hat{\mathbf{x}} = \mathbf{W}^T \mathbf{z}$ form an exact linear bijection, satisfying

$$\|\mathbf{x} - \mathbf{W}^T (\mathbf{W}\mathbf{x})\|_2 = 0 \quad (13)$$

up to machine precision. The range-matching normalization of $\mathbf{c}_{\text{LLL}}$ by $\sqrt{8}$ in Eq. (6) corresponds to left-multiplying $\mathbf{W}$ by an invertible diagonal matrix $\mathbf{D}$, so the full pipeline operator $\mathbf{DW}$ remains an exact bijection with inverse $\mathbf{W}^T \mathbf{D}^{-1}$.

The spatial-to-wavelet mapping is therefore strictly lossless, limited only by machine epsilon, so any approximation error in our pipeline stems entirely from sources outside this transform.

3.5 Network Architecture

3D U-Net Velocity Network. Our velocity network $\mathbf{v}_\theta$ is a 3D U-Net (Figure 5) that takes 16 input channels (the concatenation of the noisy wavelet coefficients $\tilde{\mathbf{x}}_t$ and the conditioning observation $\tilde{\mathbf{y}}$, each in $\mathbb{R}^{8 \times (N/2)^3}$) and predicts an 8-channel velocity in wavelet space. The encoder downsamples to a fixed $8 \times 8 \times 8$ bottleneck (2, 3, and 4 stages for 32^3 , 64^3 , and 128^3 inputs, respectively), and the decoder mirrors it with skip connections. Each level uses $N_{\text{res}} = 2$ BigGAN-style residual blocks conditioned on a Gaussian Fourier time embedding [29], with base feature dimension $n_f = 32$. The full architecture, including residual block equations, time embedding, and forward-pass pseudocode, is detailed in Appendix A.

To exploit the structure of wavelet representations, we introduce two wavelet-aware architectural modifications.

Scale-Specific Conditioning. Rather than treating all wavelet bands uniformly, we inject scale-specific information at each resolution level of the U-Net. At resolution level i , we project the 8-channel wavelet observation through a $1 \times 1 \times 1$ convolution:

$$\mathbf{z}_i = \text{Conv}_{1 \times 1 \times 1}(\tilde{\mathbf{y}}) \in \mathbb{R}^{n_f \cdot m_i \times N_i^3} \quad (14)$$

followed by adaptive average pooling to match the spatial dimensions. This conditioning is added to the hidden state after each residual block, providing direct access to scale-relevant information at each resolution.

Cross-Scale Skip Connections. To facilitate information flow between different wavelet scales during generation, we introduce additional skip connections that bridge encoder features directly to corresponding decoder levels through learned $1 \times 1 \times 1$ projections:

$$\mathbf{h}_{\text{dec}}^{(i)} = \mathbf{h}_{\text{dec}}^{(i)} + \text{Conv}_{1 \times 1 \times 1}(\mathbf{h}_{\text{enc}}^{(i)}) \quad (15)$$

These connections allow the model to maintain coherent scale relationships throughout the generation process.

3.6 Training and Inference

The complete training and inference procedures are detailed in Algorithm 1 and Algorithm 2, respectively.

Training. We use AdamW optimization with learning rate $\eta = 10^{-4}$ and a ReduceLROnPlateau scheduler with patience 5 and factor 0.5. Gradient clipping with maximum norm 1.0 ensures stable training. Data augmentation via random 3D rotations and reflections improves generalization. Each model is trained for 100 epochs, and the

Algorithm 1 Cosmo3DFlow Training

```

1: Input: Dataset  $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^M$ , velocity model  $\mathbf{v}_\theta$ , power-
   spectrum weight  $\lambda_{\text{PS}}$ 
2: Output: Trained parameters  $\theta^*$ 
3: Initialize  $\mathbf{v}_\theta$  and EMA parameters with decay 0.99
4: for epoch  $e = 1, \dots, E_{\text{max}}$  do
5:   for each batch  $(\mathbf{x}, \mathbf{y})$  do
6:     Stage 1: Wavelet encoding
7:     Sample  $t \sim \mathcal{U}(0, 1)$  and  $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
8:     Wavelet transform:  $\tilde{\mathbf{x}}_0 \leftarrow \text{DWT}(\mathbf{x}_0)$ ,  $\tilde{\mathbf{x}}_1 \leftarrow \text{DWT}(\mathbf{x})$ ,
9:      $\tilde{\mathbf{y}} \leftarrow \text{DWT}(\mathbf{y})$ 
10:    Interpolate:  $\tilde{\mathbf{x}}_t \leftarrow t \tilde{\mathbf{x}}_1 + (1 - t) \tilde{\mathbf{x}}_0$ 
11:    Stage 2: Velocity prediction
12:    Predict velocity via Algorithm 3:  $\mathbf{v}_t \leftarrow \mathbf{v}_\theta(\tilde{\mathbf{x}}_t, t, \tilde{\mathbf{y}})$ 
13:    Stage 3: Loss computation and update
14:    Flow-matching loss:  $\tilde{\mathcal{L}}_{\text{flow}} \leftarrow \|\mathbf{v}_t - (\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_0)\|_2^2$ 
15:    Predicted power spectrum:
16:     $P_{\text{pred}} \leftarrow \text{PowerSpec}(\text{IDWT}(\mathbf{v}_t))$ 
17:    Target power spectrum:
18:     $P_{\text{target}} \leftarrow \text{PowerSpec}(\text{IDWT}(\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_0))$ 
19:    Power-spectrum loss:
20:     $\mathcal{L}_{\text{PS}} \leftarrow \|\log P_{\text{pred}} - \log P_{\text{target}}\|_2^2$ 
21:    Total loss:  $\mathcal{L}_{\text{total}} \leftarrow \tilde{\mathcal{L}}_{\text{flow}} + \lambda_{\text{PS}} \mathcal{L}_{\text{PS}}$ ; update  $\theta$  by gradient
    descent; update EMA
22:   end for
23: end for
24: Return: EMA-averaged  $\theta^*$ 

```

Note: gray marks our modifications.

Algorithm 2 Cosmo3DFlow Inference: Sample Initial Conditions

```

1: Input: Observed  $\mathbf{y}$ , trained EMA model  $\mathbf{v}_{\theta^*}$ , ODE steps  $K$ 
2: Output: Initial condition  $\hat{\mathbf{x}}$ 
3: Stage 1: Wavelet encoding
4: Compute  $\tilde{\mathbf{y}} \leftarrow \text{DWT}(\mathbf{y})$ 
5: Sample  $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  in voxel space and set  $\tilde{\mathbf{x}}_0 \leftarrow \text{DWT}(\mathbf{x}_0)$ 
6: Stage 2: ODE integration
7: for  $n = 1$  to  $K$  do
8:   Set  $t \leftarrow (n - 1)/K$ 
9:   Predict velocity:  $\mathbf{v} \leftarrow \mathbf{v}_{\theta^*}(\tilde{\mathbf{x}}_{n-1}, t, \tilde{\mathbf{y}})$ 
10:  Euler step:  $\tilde{\mathbf{x}}_n \leftarrow \tilde{\mathbf{x}}_{n-1} + (1/K) \mathbf{v}$ 
11: end for
12: Stage 3: Decoding
13: Recover  $\hat{\mathbf{x}} \leftarrow \text{IDWT}(\tilde{\mathbf{x}}_K)$ 
14: Return:  $\hat{\mathbf{x}}$ 

```

best model by validation loss is selected for testing. We use batch sizes 16, 8, and 4 for the 32^3 , 64^3 , and 128^3 resolutions, respectively. All training and inference use the NVIDIA A100 GPU with 80 GB of memory.

Inference. At test time, we sample Gaussian noise in voxel space, transform to the wavelet domain, integrate the learned velocity field using the Euler method with K steps, and recover the final density field via the IDWT. The wavelet-space formulation enables accurate sampling with far fewer steps than diffusion-based approaches,

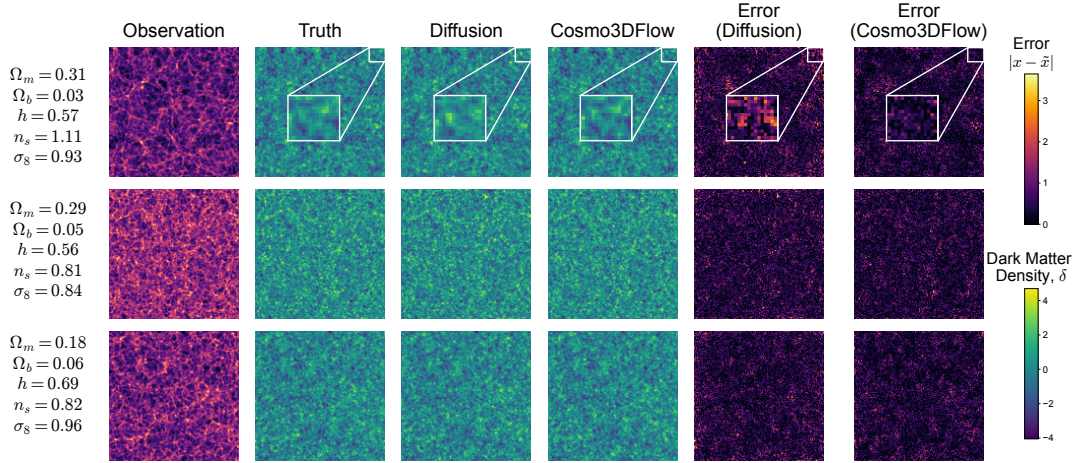


Figure 3: Qualitative comparison of reconstruction quality. We show 2D slices of 3D density fields from the Standard Latin Hypercube, reconstructing initial conditions from observations. Compared to diffusion, Cosmo3DFlow preserves sharp, small-scale structures (highlighted boxes) without blurring artifacts, yielding consistently lower errors in high-density regions.

consistent with the Euler stability lower bound on the ODE step size.

4 Experiments

4.1 Experimental Settings

Dataset. We evaluate on the following datasets from the Quijote Latin Hypercube (LH) simulations¹ [32], summarized in Table 1. All suites use a periodic box volume of $(1000 h^{-1} \text{ Mpc})^3$ with 512^3 particles. For the Standard LH and Big Sobol Sequence (BSQ) sets, the five core Λ CDM parameters are varied. In the $f_{\text{NL}}^{\text{local}}$ set, these parameters are held fixed at the fiducial cosmology ($\Omega_m = 0.3175$, $\Omega_b = 0.049$, $h = 0.6711$, $n_s = 0.9624$, $\sigma_8 = 0.834$) to isolate the impact of primordial non-Gaussianity on large-scale structure. We collect the snapshots from the publicly available source at redshifts $z = 127$ (initial conditions) and $z = 0$ (present-day universe).

Table 1: Configurations of the Quijote simulation subsets used to train and evaluate our models.

Dataset	Samples	Parameters	Description
SLH	2,000	5 core Λ CDM	Baseline cosmology
BSQ	1,000	5 core Λ CDM	Extended sampling
NG-LH	1,000	$f_{\text{NL}}^{\text{local}} \in [-300, 300]$	Non-Gaussianity

Standard LH (SLH). This suite comprises 2,000 full N -body dark matter simulations with cosmological parameters sampled uniformly via Latin-Hypercube sampling across $\Omega_m \in [0.1, 0.5]$, $\Omega_b \in [0.03, 0.07]$, $\sigma_8 \in [0.6, 1.0]$, $h \in [0.5, 0.9]$, and $n_s \in [0.8, 1.2]$. We partition the simulations into training, validation, and test sets of sizes 1,800, 100, and 100, respectively.

Big Sobol Sequence (BSQ). A large collection of N -body simulations designed for machine learning, varying the same 5 core Λ CDM

parameters as the SLH but generated using a Sobol Sequence. We use the first 1,000 samples, split 8:1:1 for training, validation, and test.

Non-Gaussian Local LH (NG-LH). 1,000 simulations exploring the impact of primordial non-Gaussianities on large-scale structure by varying $f_{\text{NL}}^{\text{local}} \in [-300, 300]$ via Latin-Hypercube sampling, split 8:1:1.

Data generation budget. Because we utilize a subset of the publicly available Quijote suite [32], no local compute budget was required to generate the training data. For context, generating the original simulation suite required approximately 35 million CPU core-hours.

Density Field Construction. The 3D matter density fields are constructed from the N -body simulation snapshots using the Pylians library [31] following [12]. For the late-time snapshots at $z = 0$, we employ the Piecewise Cubic Spline mass assignment scheme to minimize aliasing and ensure a high-fidelity representation of the non-linear cosmic web. For the initial conditions at $z = 127$, we use the Cloud-in-Cell (CIC) scheme, which is sufficient for the low-contrast fluctuations at high redshift. Each density field is computed at three resolutions (32^3 , 64^3 , and 128^3) and converted into the dimensionless overdensity field, $\delta = \rho/\bar{\rho} - 1$, to serve as the input and target for our generative models. We apply a logarithmic transformation to the $z = 0$ observation fields due to their high dynamic range. Both fields are standardized afterwards for model training.

Evaluation Metrics. Pointwise error alone is insufficient for scientific use: a model can attain low VRMSE while failing to recover the power spectrum $P(k)$, so we additionally evaluate cosmological summary statistics that the reconstructed field must respect. We consider several metrics for evaluation, each serving a different purpose. All are implemented using the Pylians library [31].

- (1) **Variance-normalized RMSE (VRMSE).** The root mean squared error (RMSE) and its normalized variants are commonly used metrics to quantify the pointwise accuracy of

¹<https://quijote-simulations.readthedocs.io/en/latest/LH.html>

Table 2: Performance on Quijote datasets. Cosmo3DFlow (Ours, shaded grey) consistently outperforms the diffusion baseline across resolutions. Bold indicates substantial improvements.

Dataset	N^3	VRMSE↓		Cross Correlation↑		Power Spectrum R^2 ↑		Transfer Function↑	
		Ours	Diffusion	Ours	Diffusion	Ours	Diffusion	Ours	Diffusion
Standard Latin Hypercube	128 ³	0.50	0.63	0.88	0.82	0.99	0.70	0.99	0.80
	64 ³	0.47	0.68	0.92	0.89	0.98	0.59	0.98	0.59
	32 ³	0.25	0.82	0.98	0.85	0.97	0.48	0.96	0.48
Big Sobol Sequence	128 ³	0.62	0.64	0.80	0.79	0.99	0.84	0.95	0.88
	64 ³	0.53	0.65	0.88	0.88	0.98	0.83	0.94	0.81
	32 ³	0.37	0.79	0.95	0.85	0.95	0.48	0.94	0.71
Non-Gauss (f_{NL} LH)	128 ³	0.56	0.59	0.86	0.83	1.00	1.00	0.98	0.98
	64 ³	0.47	0.57	0.93	0.89	1.00	1.00	0.99	0.99
	32 ³	0.31	0.67	0.97	0.87	1.00	0.98	0.99	0.98

simulations [20, 24]. Formally, for two spatial fields $\mathbf{a}$ and $\mathbf{b}$, the VRMSE is defined as follows, where $\langle \cdot \rangle$ denotes the spatial mean operator, and $\epsilon = 10^{-6}$ is a numerical stability term:

$$\text{VRMSE}(\mathbf{a}, \mathbf{b}) = \sqrt{\frac{\langle (\mathbf{a} - \mathbf{b})^2 \rangle}{\langle (\mathbf{a} - \langle \mathbf{a} \rangle)^2 \rangle + \epsilon}} \quad (16)$$

- (2) **Power Spectrum R^2 -score.** The predicted dark matter density distribution should remain similar to the ground truth. Following [23], we calculate the R^2 -score between the target 3D density field and the predicted samples. This shows how the model performs on the power spectrum.
- (3) **Cross-Correlation, $C(k)$.** This metric measures the scale-dependent correlation between reconstructed and true density fields, providing a metric insensitive to overall amplitude normalization [10, 18]. It is computed using the following, where $P_{11}(k)$ and $P_{22}(k)$ denote the power spectra of the sample and truth, respectively:

$$C(k) = \frac{P_{12}(k)}{\sqrt{P_{11}(k) \cdot P_{22}(k)}} \quad (17)$$

- (4) **Transfer Function, $T(k)$.** The transfer function $T(k)$ characterizes the scale-dependent fidelity of reconstruction by measuring the ratio of recovered to true power at each spatial scale. We define the transfer function as: $T(k) = \sqrt{P_{\text{sample}}(k)/P_{\text{target}}(k)}$.

4.2 Main Results

Baseline. We use the score-based diffusion model from Legin et al. [12] as the baseline generative model, as it represents the state-of-the-art on our target problem. We refer to it as the “diffusion” baseline. The U-Net parameters are kept the same as in our model to obtain a fixed latent dimension ($8 \times 8 \times 8$) across resolutions; hence, the compression ratio matches across resolutions. This ensures a fair comparison between the two models.

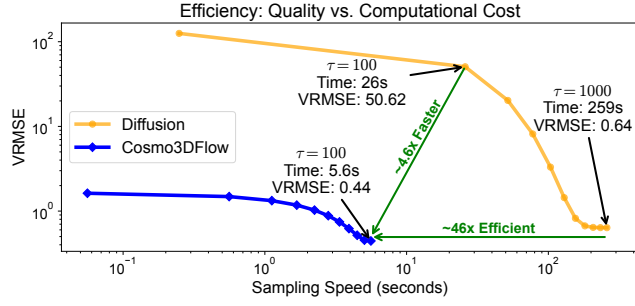
Comparison to the diffusion baseline. We assess our method against the diffusion baseline along four axes. Cosmo3DFlow preserves sharp small-scale structure without the blurring seen in the

baseline (Figure 3). Quantitatively, it reaches matching reconstruction quality with substantially fewer sampling steps (Figure 4a), converges faster in VRMSE (Figure 4b), and aligns near-perfectly with the truth on power spectrum, cross-correlation, and transfer function (Figure 4c). As summarized in Table 2, Cosmo3DFlow consistently outperforms the diffusion baseline across all datasets and resolutions. At 128³ on the Standard Latin Hypercube, it achieves a VRMSE of 0.50 vs. 0.63, cross-correlation of 0.88 vs. 0.82, and a power spectrum R^2 -score of 0.99 vs. 0.70. These gains are consistent across the Big Sobol Sequence and Non-Gaussian datasets, demonstrating robustness to varying cosmological parameters and primordial non-Gaussianity. This baseline uses a stochastic differential equation (SDE) for diffusion sampling. To ensure an equitable comparison, we further isolate the contribution of deterministic ODE sampling from our wavelet-space formulation and additionally compare Cosmo3DFlow (ODE) against the deterministic Probability Flow ODE (PF-ODE) variant of the diffusion baseline using the Euler method, as shown in Table 9 of the Appendix. As reported on the Quijote Standard Latin Hypercube, Cosmo3DFlow still maintains a sizable margin in speed and quality.

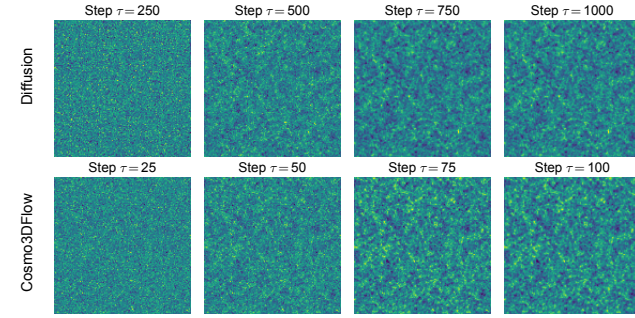
Appendix B reports posterior calibration and diversity for our model.

4.3 Generalization to Realistic Observational Tracers

To demonstrate that the Cosmo3DFlow model works not only on dark-matter density fields, but also on realistic observational inputs, we also train and evaluate on the Quijote Standard *halo dataset* at 128³, where halos act as sparse, biased, and noisy tracers that mimic actual telescope survey data (e.g., DESI or Euclid surveys). Because the true initial conditions of the real universe are unobservable, halos—the gravitationally bound structures that host galaxies and galaxy clusters—provide the most faithful ground-truth signal available. Prior work [7, 21] follows the same convention of training and evaluating on simulated datasets. As shown in Table 3, Cosmo3DFlow remains robust under this noisier and sparser conditioning, outperforming the diffusion baseline on all four metrics.



(a) Efficiency vs. quality.



(b) Sample generation from Gaussian noise.

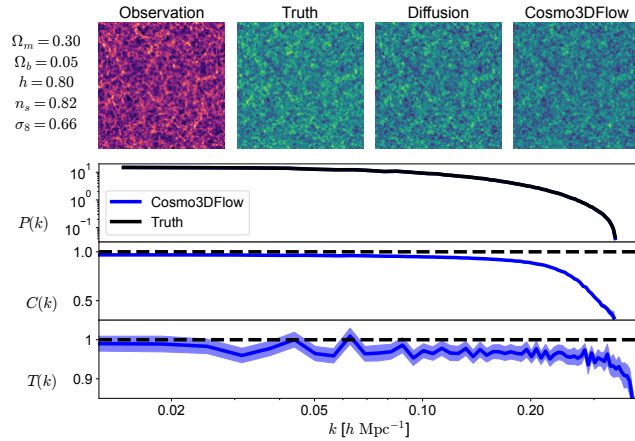
(c) Quantitative evaluation on the Standard LH, resolution 128^3 .

Figure 4: Quantitative Results. (a) Cosmo3DFlow achieves 4.6× lower per-step cost. (b) VRMSE progression shows significantly faster convergence. (c) Near-perfect alignment with the truth across power spectrum, cross-correlation, and transfer function. Note the logarithmic scale on the horizontal axis.

Table 3: Evaluation on the Quijote Standard halo dataset (128^3), where halos act as sparse, biased tracers of the underlying density. Bold indicates the best in each column.

Model	VRMSE↓	Cross Correlation↑	R^2 ↑	Transfer Function↑
Diffusion	1.08	0.55	0.85	0.85
Ours	0.93	0.56	0.90	0.91

4.4 Ablation Study

Component ablation. We evaluate the impact of our added components on model performance at 32^3 resolution using the Standard Latin Hypercube set. For a fair comparison, Voxel FM (without wavelets) utilizes an additional downsampling/upsampling layer to maintain the same compression ratio with a latent dimension $8 \times 8 \times 8$ as the wavelet-based configurations. Table 4 shows that Voxel FM already achieves good performance. Naively applying wavelet flow matching, without our scale-aware architectural interventions, yields a worse VRMSE than Voxel FM (0.34 vs. 0.26), indicating that simply stacking wavelets onto flow matching is insufficient on highly sparse cosmological data. Adding Scale Conditioning and Cross-Scale Skips bridges the gap between the spatial and wavelet domains, providing the necessary context to resolve features across different scales, and the power-spectrum weighting balances reconstruction error across wavenumbers. We omitted the cross-correlation because all 32^3 models yielded approximately 0.98.

Table 4: Ablation study of model components evaluated on the Standard Latin Hypercube, showing that the added components have a positive impact on the overall performance. Bold indicates the best in each column.

Components				Evaluation Metrics		
Wave	Cond	Skip	$P(k)$	VRMSE↓	R^2 ↑	$T(k)$ ↑
×	×	×	×	0.26	0.95	0.94
✓	×	×	×	0.34	0.95	0.95
✓	✓	✓	×	0.27	0.96	0.96
✓	✓	✓	0.01	0.25	0.97	0.96

Components: Wave = wavelet decomposition; Cond = scale conditioning; Skip = cross-scale skips; $P(k)$ = power-spectrum weighting. *Metrics:* VRMSE, power-spectrum R^2 , and transfer function $T(k)$.

4.5 Complexity and Scaling Analysis

Memory and runtime efficiency. The Cosmo3DFlow framework performs computations within a decomposed coefficient space rather than the high-resolution spatial domain. As shown in Table 5, this mitigates the cubic scaling bottlenecks inherent in 128^3 volumes, achieving a 68.2% reduction in peak memory compared to standard flow matching and a 55.7% reduction relative to the diffusion baseline at 128^3 . The wavelet decomposition thus preserves structural fidelity while lowering computational overhead; the per-factor speed breakdown is detailed below.

Table 5: Wavelet-space contribution: Sampling efficiency of Cosmo3DFlow vs. standard voxel-space flow matching (Voxel FM) and diffusion.

N^3	Metric	Diffusion	Voxel FM	Ours	Gain
32^3	Time (s)	10.8	1.32	1.06	10.2×
	Memory (GB)	0.22	0.37	0.21	4.5%
64^3	Time (s)	39.2	3.51	1.52	25.8×
	Memory (GB)	0.80	0.81	0.37	53.8%
128^3	Time (s)	259	23.7	5.60	46.2×
	Memory (GB)	3.93	5.47	1.74	55.7%

Gain: relative improvement against Diffusion (Speedup for Time, % Less for Memory).

Table 6: Per-component theoretical complexity of Cosmo3DFlow (Ours) vs. the diffusion baseline.

Component	Diffusion	Ours
Asymptotic scaling	$O(N^3)$	$O(N^3)$
Spatial domain	N^3	$N^3/8$
Integration steps	$K'=1,000$	$K=100$
Sampling complexity	$O(K'N^3C)$	$O(KN^3C/8)$

Symbols: N = grid size; C = channels; K, K' = sampling step counts.

Table 7: Cosmo3DFlow at 256^3 on Standard Latin Hypercube.

Setup	VRMSE↓	$C(k)$ ↑	R^2 ↑	$T(k)$ ↑
128^3	0.50	0.88	0.99	0.99
256^3	0.85	0.55	0.97	0.95
256^3 +TL [†]	0.68	0.72	0.96	0.92

Metrics: VRMSE, cross-correlation $C(k)$, power spectrum R^2 , and transfer function $T(k)$. [†] denotes transfer learning (TL) from the 128^3 checkpoint.

Theoretical ceiling vs. practical efficiency. Both approaches share the same $O(N^3)$ asymptotic upper bound; our model’s acceleration comes from reducing the constant factors. Composing the per-step $8\times$ spatial-domain factor in Table 6 with the $10\times$ reduction in integration steps yields an $\approx 80\times$ theoretical end-to-end efficiency gain, against the $46\times$ empirical wall-clock observed at 128^3 (Table 5). The remaining gap is dominated by 3D convolutions being memory-bandwidth-bound rather than FLOPs-bound on the target hardware, together with fixed costs that do not shrink with N (DWT/IDWT, time embedding, kernel launches); accordingly the observed acceleration grows from $10.2\times$ at 32^3 to $25.8\times$ at 64^3 to $46.2\times$ at 128^3 , converging toward the theoretical ceiling as N increases.

Scaling to 256^3 . Our model extends naturally to the 256^3 full volume, a regime in which the diffusion baseline runs out of memory (projected ~ 150 GB VRAM for training, extrapolated from the 128^3 row of Table 5). Training Cosmo3DFlow at 256^3 requires 23 GB of VRAM (batch size 2) and 40 GPU-hours on a single A100 for

100 epochs over 1,800 samples. At inference, each posterior sample takes about 41 seconds and 13 GB of VRAM. Transfer learning using the 128^3 checkpoint substantially accelerates convergence and improves accuracy, as shown in Table 7. The diffusion baseline cannot be trained at 256^3 due to out-of-memory errors.

5 Conclusion

We introduced Cosmo3DFlow, a generative framework combining flow matching with wavelet-space modeling for high-dimensional inverse problems. When density fields are transformed into the wavelet domain, voids collapse into near-zero high-frequency coefficients while informative structures are preserved compactly—a physically informed exploitation of *wavelet sparsity*. This achieves up to $46\times$ **faster** sampling than diffusion baselines at 128^3 resolution, as wavelet decomposition naturally aligns with multi-scale cosmological structure, enabling sparse velocity fields that require far fewer ODE integration steps.

As cosmological simulations scale to larger volumes and finer resolutions (512^3 , 1024^3), the voxel void fraction will remain constant or increase (as we resolve smaller voids), but the computational cost of processing it on a grid will scale cubically (N^3). Wavelet sparsity mitigates this scaling bottleneck. It ensures that computational resources are allocated to the structure of the universe—the mass, the energy, and the information—rather than its emptiness. This shift from spatial to spectral modeling represents not just an optimization, but a necessary evolution for the next generation of high-dimensional astrophysical inference.

This work provides valuable insights into mapping sparse spatial structures to the frequency domain, and the proposed Wavelet Flow Matching paradigm enables more stable, efficient ODE solvers with larger steps while preserving fine detail. We believe the core algorithmic innovation of this paper—translating spatial emptiness into spectral sparsity—can be generalized to other fields such as medical imaging (MRI/CT), by enabling high-resolution synthesis; computational fluid dynamics, by focusing resources on turbulent regions; and meteorology, by improving global forecasting resolution while bypassing grid-scale computational limits.

Limitations and Ethical Considerations

While our method proves robust on both dark matter density fields and halo catalogs derived from N -body simulations, full hydrodynamical simulation suites (incorporating baryonic physics) and alternative wavelet families remain to be explored. The synthetic simulation data poses no privacy or ethical concerns.

GenAI Disclosure

Large Language Models (LLMs) were used for the initial drafting of figures and text polishing.

Acknowledgments

We acknowledge support from the National Science Foundation under Cooperative Agreement 2421782, the Simons Foundation through award MPS-AI-00010515 and NSF Simons Cosmic AI Seed Grant AWD-006703. We are grateful for the resources and support provided by UVA Research Computing.

References

- [1] J. M. Bardeen, J. R. Bond, N. Kaiser, and A. S. Szalay. 1986. The Statistics of Peaks of Gaussian Random Fields. *The Astrophysical Journal* 304 (May 1986), 15. doi:10.1086/164143
- [2] Atrideb Chatterjee and Francisco Villaescusa-Navarro. 2025. Cosmology from Point Clouds with Dark Matter Halos from the Quijote Simulations. *The Astrophysical Journal* 985, 1 (May 2025), 132. doi:10.3847/1538-4357/adc99d
- [3] Scott Dodelson and Fabian Schmidt. 2021. *Modern Cosmology*. Elsevier, London. doi:10.1016/C2017-0-01943-2
- [4] Paul Friedrich, Julia Wolle, Florentin Bieder, Alicia Durrer, and Philippe C. Cattin. 2025. WDM: 3D Wavelet Diffusion Models for High-Resolution Medical Image Synthesis. In *Deep Generative Models*, Anirban Mukhopadhyay, Ilkay Okuz, Sandy Engelhardt, Dorit Mehrhof, and Yixuan Yuan (Eds.). Springer Nature Switzerland, Cham, 11–21. doi:10.1007/978-3-031-72744-3_2
- [5] Ernst Hairer, Gerhard Wanner, and Syvert P. Nørsett. 1993. *Solving Ordinary Differential Equations I*. Springer Series in Computational Mathematics, Vol. 8. Springer, Berlin, Heidelberg. doi:10.1007/978-3-540-78862-1
- [6] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., Red Hook, NY, USA, 6840–6851. https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bfc8584af0d967f1ab10179ca4b-Paper.pdf
- [7] Teresa Huang, Richard Stiskalek, Jun-Young Lee, Adrian Bayer, Charles Margossian, Christian Kragh Jespersen, Lucia Perez, Lawrence Saul, and Francisco Villaescusa. 2025. CosmoBench: A Multiscale, Multiview, Multitask Cosmology Benchmark for Geometric Deep Learning. In *Advances in Neural Information Processing Systems*, D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (Eds.), Vol. 38. Curran Associates, Inc., Red Hook, NY, USA, 40 pages. https://proceedings.neurips.cc/paper_files/paper/2025/file/977f2451cb616e89dbb819434a932e7e-Paper-Datasets_and_Benchmarks_Track.pdf
- [8] Dragan Huterer. 2023. *A Course in Cosmology: From Theory to Practice*. Cambridge University Press, Cambridge.
- [9] J. Jasche and G. Lavaux. 2019. Physical Bayesian Modelling of the Non-Linear Matter Distribution: New Insights into the Nearby Universe. *Astronomy & Astrophysics* 625 (May 2019), A64. doi:10.1051/0004-6361/201833710
- [10] Jens Jasche and Benjamin D. Wandelt. 2013. Bayesian Physical Reconstruction of Initial Conditions from Large-Scale Structure Surveys. *Monthly Notices of the Royal Astronomical Society* 432, 2 (June 2013), 894–913. doi:10.1093/mnras/stt449
- [11] Sidharth Kannan, Tian Qiu, Carolina Cuesta-Lazaro, and Haewon Jeong. 2025. CosmoFlow: Scale-Aware Representation Learning for Cosmology with Flow Matching. arXiv:2507.11842 [astro-ph.CO]
- [12] Ronan Legin, Matthew Ho, Pablo Lemos, Laurence Perreault-Levasseur, Shirley Ho, Yashar Hezaveh, and Benjamin Wandelt. 2024. Posterior Sampling of the Initial Conditions of the Universe from Non-Linear Large Scale Structures Using Score-Based Generative Models. *Monthly Notices of the Royal Astronomical Society: Letters* 527, 1 (Jan. 2024), L173–L178. doi:10.1093/mnras/lsad152
- [13] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2023. Flow Matching for Generative Modeling. arXiv:2210.02747 [cs.LG]
- [14] Xingchao Liu, Chengyue Gong, and Qiang Liu. 2022. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. arXiv:2209.03003 [cs.LG]
- [15] M. Mebratu and W.L.K. Wu. 2026. Wavelet Flow for Extragalactic Foreground Simulations. *Journal of Cosmology and Astroparticle Physics* 2026, 01 (Jan. 2026), 011. doi:10.1088/1475-7516/2026/01/011
- [16] Ofer Metuki, Noam I. Libeskind, Yehuda Hoffman, Robert A. Crain, and Tom Theuns. 2015. Galaxy Properties and the Cosmic Web in Simulations. *Monthly Notices of the Royal Astronomical Society* 446, 2 (Jan. 2015), 1458–1468. doi:10.1093/mnras/stu2166
- [17] Satvik Mishra, Roberto Trotta, and Matteo Viel. 2026. Cosmo-FOLD: Fast Generation and Upscaling of Field-Level Cosmological Maps with Overlap Latent Diffusion. arXiv:2601.14377 [astro-ph.CO]
- [18] C. Modi, F. Lanusse, and U. Seljak. 2021. FlowPM: Distributed TensorFlow Implementation of the FastPM Cosmological N-body Solver. *Astronomy and Computing* 37 (Oct. 2021), 100505. doi:10.1016/j.ascom.2021.100505
- [19] Chirag Modi, François Lanusse, Uroš Seljak, David N. Spergel, and Laurence Perreault-Levasseur. 2021. CosmicRIM: Reconstructing Early Universe by Combining Differentiable Simulations with Recurrent Inference Machines. arXiv:2104.12864 [astro-ph.CO]
- [20] Ruben Ohana, Michael McCabe, Lucas Meyer, Rudy Morel, Fruzsina J. Agocs, Miguel Benítez, Marsha Berger, Blakesley Burkhart, Stuart B. Dalziel, Drummond B. Fielding, Daniel Fortunato, Jared A. Goldberg, Keiya Hirashima, Yan-Fei Jiang, Rich R. Kerswell, Suryanarayana Maddu, Jonah Miller, Payel Mukhopadhyay, Stefan S. Nixon, Jeff Shen, Romain Watteaux, Bruno R. Blacard, François Rozet, Liam Parker, Miles Cranmer, and Shirley Ho. 2024. The Well: A Large-Scale Collection of Diverse Physics Simulations for Machine Learning. *Advances in Neural Information Processing Systems* 37 (Dec. 2024), 44989–45037. doi:10.52202/079017-1430
- [21] Liam Parker, Adrian E. Bayer, and Uroš Seljak. 2025. Initial Conditions from Galaxies: Machine-Learning Subgrid Correction to Standard Reconstruction. *Journal of Cosmology and Astroparticle Physics* 2025, 09 (Sept. 2025), 039. doi:10.1088/1475-7516/2025/09/039
- [22] Hao Phung, Quan Dao, and Anh Tran. 2023. Wavelet Diffusion Models Are Fast and Scalable Image Generators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Vancouver, BC, Canada, 10199–10208. https://openaccess.thecvf.com/content/CVPR2023/papers/Phung_Wavelet_Diffusion_Models_Are_Fast_and_Scalable_Image_Generators_CVPR_2023_paper.pdf
- [23] Julieth K Riveros, Paola A Saavedra, Héctor J Hortúa, Jorge Enrique García-Farieta, and Ivan Olier. 2025. Conditional Diffusion-Flow Models for Generating 3D Cosmic Density Fields: Applications to f(R) Cosmologies. *Machine Learning: Science and Technology* 6, 3 (Aug. 2025), 035031. doi:10.1088/2632-2153/adf8b1
- [24] François Rozet, Ruben Ohana, Michael McCabe, Gilles Louppe, François Lanusse, and Shirley Ho. 2026. Lost in Latent Space: An Empirical Study of Latent Diffusion Models for Physics Emulation. *Advances in Neural Information Processing Systems* 38 (April 2026), 134612–134657. https://proceedings.neurips.cc/paper_files/paper/2025/file/c3b88408bb209dca87dad596ecf88bba-Paper-Conference.pdf
- [25] Hamees Sayed, Pranath Reddy, Michael W. Toomey, and Sergei Gleyzer. 2025. FlowLensing: Simulating Gravitational Lensing with Flow Matching. arXiv:2510.07878 [astro-ph.IM]
- [26] Christopher J Shallue and Daniel J Eisenstein. 2023. Reconstructing Cosmological Initial Conditions from Late-Time Structure with Convolutional Neural Networks. *Monthly Notices of the Royal Astronomical Society* 520, 4 (April 2023), 6256–6267. doi:10.1093/mnras/stad528
- [27] Luigi Sigillo, Shengfeng He, and Danilo Comminiello. 2026. Latent Wavelet Diffusion For Ultra-High-Resolution Image Synthesis. arXiv:2506.00433 [cs.CV]
- [28] Yang Song and Stefano Ermon. 2020. Improved Techniques for Training Score-Based Generative Models. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., Red Hook, NY, USA, 12438–12448. https://proceedings.neurips.cc/paper_files/paper/2020/file/92c3b916311a5517d9290576e3ea37ad-Paper.pdf
- [29] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. 2020. Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., Red Hook, NY, USA, 7537–7547. https://proceedings.neurips.cc/paper_files/paper/2020/file/55053683268957697aa39fba6f231c68-Paper.pdf
- [30] Georgios Valogiannis, Francisco Villaescusa-Navarro, and Marco Baldi. 2024. Towards Unveiling the Large-Scale Nature of Gravity with the Wavelet Scattering Transform. *Journal of Cosmology and Astroparticle Physics* 2024, 11 (Nov. 2024), 061. doi:10.1088/1475-7516/2024/11/061
- [31] Francisco Villaescusa-Navarro. 2018. Pylians: Python Libraries for the Analysis of Numerical Simulations. *Astrophysics Source Code Library*, record ascl:1811.008. ascl:1811.008
- [32] Francisco Villaescusa-Navarro, ChangHoon Hahn, Elena Massara, Arka Banerjee, Ana Maria Delgado, Doogesh Kodi Ramanah, Tom Charnock, Elena Giusarma, Yin Li, Erwan Allys, Antoine Brochard, Cora Uhlemann, Chi-Ting Chiang, Siyu He, Alice Pisani, Andrej Obuljen, Yu Feng, Emanuele Castorina, Gabriella Contardo, Christina D. Kreisch, Andrina Nicola, Justin Alsing, Roman Scoccimarro, Licia Verde, Matteo Viel, Shirley Ho, Stéphane Mallat, Benjamin Wandelt, and David N. Spergel. 2020. The Quijote Simulations. *The Astrophysical Journal Supplement Series* 250, 1 (Aug. 2020), 2. doi:10.3847/1538-4365/ab9d82
- [33] Simon D. M. White and Carlos S. Frenk. 1991. Galaxy Formation through Hierarchical Clustering. *The Astrophysical Journal* 379 (Sept. 1991), 52. doi:10.1086/170483
- [34] Maksim Zhdanov, Max Welling, and Jan-Willem Van De Meent. 2025. Erwin: A Tree-based Hierarchical Transformer for Large-scale Physical Systems. In *Proceedings of the 42nd International Conference on Machine Learning*. PMLR, Vancouver, BC, Canada, 78023–78040. <https://proceedings.mlr.press/v267/zhdanov25a.html>

A 3D U-Net Architecture Details

Figure 5 illustrates the detailed architecture of our 3D U-Net velocity network. The network follows an encoder-decoder structure specifically designed for volumetric cosmological data processing in wavelet space. We provide more details in Algorithm 3.

Input Processing. The network receives a 16-channel input by concatenating noisy wavelet coefficients $\tilde{\mathbf{x}}_t \in \mathbb{R}^{8 \times (N/2)^3}$ with the conditioning observation $\tilde{\mathbf{y}} \in \mathbb{R}^{8 \times (N/2)^3}$. An initial $3 \times 3 \times 3$ convolution projects this to the base feature dimension $n_f = 32$.

Algorithm 3 U-Net Forward Pass with Scale Conditioning and Cross-Scale Skips

```

1: Input: Noisy wavelet coefficients  $\tilde{\mathbf{x}}_t$ , flow time  $t$ , conditioning observation  $\tilde{\mathbf{y}}$ 
2: Output: Predicted wavelet-space velocity  $\mathbf{v}_t$ 
3: Stack  $[\tilde{\mathbf{x}}_t; \tilde{\mathbf{y}}]$  as input; encode  $t$  to embedding  $\mathbf{e}_t$ ; project to feature space
4: Stage 1: Encoder with scale conditioning
5: for each encoder level  $i = 0$  to  $\mathcal{L}-1$  do
6:   Apply residual blocks guided by  $\mathbf{e}_t$ ; cache as U-Net skip
7:   Scale conditioning: inject  $\tilde{\mathbf{y}}$  at level- $i$  resolution
8:   Cross-scale skip: snapshot encoder features  $\mathbf{s}_i$ 
9:   Downsample to next coarser resolution (if  $i < \mathcal{L}-1$ )
10: end for
11: Stage 2: Bottleneck
12: Integrate global structure at the coarsest resolution
13: Stage 3: Decoder with cross-scale fusion
14: for each decoder level  $i = \mathcal{L}-1$  down to 0 do
15:   Fuse U-Net skip; apply residual blocks guided by  $\mathbf{e}_t$ 
16:   Cross-scale skip: add encoder snapshot  $\mathbf{s}_i$ 
17:   Upsample to next finer resolution (if  $i > 0$ )
18: end for
19: Project to 8-channel wavelet velocity  $\mathbf{v}_t$  via GroupNorm, SiLU, and  $3 \times 3 \times 3$  convolution
20: Return:  $\mathbf{v}_t$ 

```

Encoder Structure. The encoder consists of 2, 3, and 4 downsampling layers for the image resolutions 32, 64, and 128, respectively. Each level contains $N_{\text{res}} = 2$ residual blocks that incorporate time conditioning through additive feature modulation. Spatial downsampling between levels is performed via strided average pooling with a factor of 2.

Residual Block. The residual blocks employ the BigGAN-style architecture:

$$\mathbf{h}' = \text{Conv}(\text{SiLU}(\text{GN}(\mathbf{h}))) + \text{Dense}(\text{SiLU}(\mathbf{e}_t)) \quad (18)$$

$$\mathbf{h}'' = \text{Conv}(\text{Dropout}(\text{SiLU}(\text{GN}(\mathbf{h}')))) \quad (19)$$

$$\text{ResBlock}(\mathbf{h}, \mathbf{e}_t) = (\mathbf{h}'' + \text{Skip}(\mathbf{h})) / \sqrt{2} \quad (20)$$

where GN denotes Group Normalization with $\min(c/4, 32)$ groups (c being the number of channels), and Skip denotes a $1 \times 1 \times 1$ convolution when channel dimensions change.

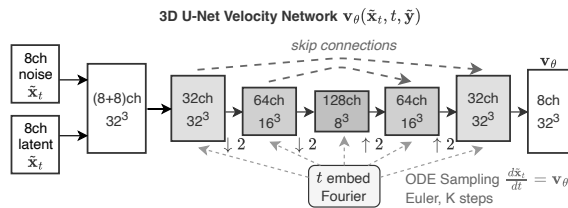


Figure 5: 3D U-Net velocity network at 64^3 . It predicts wavelet-space velocity from noisy coefficients and conditioning. The architecture features time-embeddings, skip connections, and an 8^3 latent bottleneck. See Algorithm 3 for details.

Bottleneck. At the coarsest resolution, the network processes features at the fixed 8^3 bottleneck resolution with 256 channels, capturing global context essential for coherent large-scale structure generation.

Decoder Structure. The decoder symmetrically mirrors the encoder, with each level receiving skip connections from the corresponding encoder stage. Features are upsampled via nearest-neighbor interpolation followed by convolution, then concatenated with skip features before processing through residual blocks. This design preserves fine-grained spatial information while incorporating global context.

Time Conditioning. The continuous time variable $t \in [0, 1]$ is embedded using Gaussian Fourier features:

$$\gamma(t) = [\sin(2\pi \mathbf{W}t), \cos(2\pi \mathbf{W}t)] \in \mathbb{R}^{2d} \quad (21)$$

where $\mathbf{W} \in \mathbb{R}^d$ are learnable frequency parameters initialized from $\mathcal{N}(0, \sigma^2)$ with $\sigma = 16$. This embedding is projected through two dense layers with SiLU activations to produce $\mathbf{e}_t \in \mathbb{R}^{4n_f} = \mathbb{R}^{128}$, which is injected into each residual block via a dense projection added to the intermediate features.

Output Layer. The final output is produced by Group Normalization, SiLU activation, and a $3 \times 3 \times 3$ convolution projecting to 8 output channels (predicted wavelet-space velocity).

B Posterior Diversity, Calibration, and Uncertainty

To address potential concerns about mode-seeking behavior in our conditional generator, we evaluate the Cosmo3DFlow posterior using three complementary metrics across three different inference noise scales (by default, $0.1 \times$ Gaussian noise is added to the input condition for all models):

- **Prediction Interval Coverage Probability (PICP):** the fraction of spatial locations at which the ground truth falls within the predicted credible interval.
- **Relative sample variance:** the sample variance of the generated posterior normalized by the variance of the true field.

Table 8: Posterior metrics across resolutions and noise scales. Coverage (PICP) remains high, while variance scales appropriately with noise level and finer structures (high k).

N^3	Noise	PICP	Relative Sample Variance	Variance		
				low k	mid k	high k
32^3	0.1	0.97	0.12	0.04	0.05	0.10
	0.2	0.99	0.32	0.08	0.09	0.21
	0.3	0.98	0.56	0.14	0.16	0.35
64^3	0.1	0.93	0.13	0.05	0.06	0.16
	0.2	0.95	0.23	0.07	0.09	0.26
	0.3	0.95	0.38	0.12	0.14	0.40
128^3	0.1	0.87	0.15	0.07	0.08	0.30
	0.2	0.83	0.17	0.08	0.09	0.32
	0.3	0.87	0.23	0.10	0.10	0.40

Table 9: Sampler contribution: Sampling speed and quality on the Standard Latin Hypercube. Bold indicates the best in each column within each resolution block.

N^3	Method	Efficiency		Quality			
		Memory (GB)	Time (s)	VRMSE↓	$C(k)↑$	$R^2↑$	$T(k)↑$
32^3	Diffusion (SDE)	0.22	10.8	0.82	0.85	0.48	0.48
	Diffusion (PF-ODE)	0.22	1.44	0.85	0.84	0.57	0.70
	Cosmo3DFlow (ODE)	0.21	1.06	0.25	0.98	0.97	0.96
64^3	Diffusion (SDE)	0.80	39.2	0.68	0.89	0.59	0.59
	Diffusion (PF-ODE)	0.80	3.96	0.65	0.89	0.69	0.79
	Cosmo3DFlow (ODE)	0.37	1.52	0.47	0.92	0.98	0.98
128^3	Diffusion (SDE)	3.93	259	0.63	0.82	0.70	0.80
	Diffusion (PF-ODE)	3.93	23.9	0.62	0.85	0.46	0.84
	Cosmo3DFlow (ODE)	1.74	5.60	0.50	0.88	0.99	0.99

Samplers: SDE = standard stochastic diffusion sampler (1,000 steps); PF-ODE = deterministic probability-flow variant (100 steps); Cosmo3DFlow = 100 steps.

Table 10: Notation and key definitions used.

Symbol	Definition	Symbol	Definition
<i>Problem Setup</i>			
$\mathbf{x}$	Initial conditions ($z=127$)	$\mathbf{y}$	Observations ($z=0$)
$\hat{\mathbf{x}}$	Recovered initial-condition field	z	Cosmological redshift
N	Grid resolution per dim	$p(\mathbf{x} \mathbf{y})$	Posterior of initial conds
δ	Density contrast $\rho/\bar{\rho} - 1$	$\rho, \bar{\rho}$	Density and mean density
<i>Flow Matching</i>			
t	Flow time $\in [0, 1]$	$\mathbf{x}_0$	Prior (Gaussian) sample
$\mathbf{x}_1$	Target data sample	$\mathbf{x}_t$	$t\mathbf{x}_1 + (1-t)\mathbf{x}_0$
$\mathbf{u}_t$	Target velocity $\mathbf{x}_1 - \mathbf{x}_0$	$\mathbf{v}_\theta$	Learned velocity network
K, K'	ODE integration step counts	θ	Network parameters
Δt	ODE step size	$\text{Lip}(\mathbf{v}_\theta)$	Lipschitz constant of $\mathbf{v}_\theta$
<i>Wavelet Transform</i>			
DWT	Discrete Wavelet Transform	IDWT	Inverse DWT
$\tilde{\mathbf{x}}, \tilde{\mathbf{y}}$	Wavelet-transformed fields	$\mathbf{c}_{LLL}$	Approximation coefficients
$\mathbf{c}_{LLH}, \dots, \mathbf{c}_{HHH}$	7 detail bands	L, H	Low/high-pass 1D filters
<i>Training Objectives</i>			
$\mathcal{L}_{\text{flow}}$	Voxel-space FM loss	$\tilde{\mathcal{L}}_{\text{flow}}$	Wavelet-space FM loss
$\mathcal{L}_{\text{PS}}$	Power spectrum loss	λ_{PS}	Power spectrum weight
$\mathcal{L}_{\text{total}}$	Combined training objective		
<i>Evaluation Metrics</i>			
VRMSE	Variance-normalized RMSE	$k, \mathbf{k}$	Wavenumber (scalar/vector)
$P(k)$	Isotropic power spectrum	$C(k)$	Cross-correlation at k
$T(k)$	Transfer function	PICP	Pred. Interval Coverage Prob.
ϵ	Numerical stabilizer	$\langle \cdot \rangle$	Spatial mean operator

- *Diversity power spectrum:* a scale-dependent Fourier-space measure of how sample variance, normalized by the true power, behaves as a function of wavenumber k . We partition the power spectrum into three equal bands in $\log k$ space (large, medium, and small structures).

Results are reported in Table 8. Across resolutions and noise scales, Cosmo3DFlow exhibits excellent coverage (83–99%) and healthy sample diversity (12–56%). The variance is consistently 2.5–4.3× larger at small scales (high k) than at large scales, which is physically expected: small-scale primordial features are largely erased by non-linear gravitational collapse and are therefore intrinsically harder to constrain from the $z = 0$ observation, so the posterior correctly expresses larger uncertainty.

C Sampling Speed: SDE vs. PF-ODE vs. Cosmo3DFlow

To isolate the contribution of deterministic ODE sampling from our wavelet-space formulation, we additionally compare Cosmo3DFlow against the deterministic Probability Flow ODE (PF-ODE) variant of the diffusion baseline using the Euler method. To ensure a fair comparison, both our ODE and the PF-ODE use 100 sampling steps; only the diffusion SDE retains its standard 1,000 steps. Results on the Quijote Standard Latin Hypercube are reported in Table 9. PF-ODE substantially accelerates diffusion sampling, but Cosmo3DFlow still maintains a sizable margin in speed and quality. At 128^3 , our model outperforms across all metrics and remains over 4× faster than PF-ODE (5.60 s vs. 23.9 s) while using less than half the GPU memory.

D Notations

Table 10 summarizes the notation and key definitions used throughout this paper.