A META-ALGORITHM FOR AB INITIO RECONSTRUCTION OF COMPLEX MIXTURES IN CRYO-EM

Anonymous authors
Paper under double-blind review

ABSTRACT

We describe a systematic approach for spawning and aggregating multi-class cryo-EM reconstruction jobs. This approach formalizes standard ad hoc strategies of iterative classification and filtering typically used by practitioners to sort impure, heterogeneous samples. To our knowledge, this is the first method that can successfully perform *ab initio* reconstruction on datasets containing dozens of distinct species. We obtain 97% accuracy on *ab initio* reconstruction of a 45-class subset of Tomotwin-100, 75% accuracy on the full Tomotwin-100 dataset, and demonstrate recovery of ribosomal assembly states from an unfiltered experimental cryo-EM dataset. Our approach's capability scales with compute and lays the foundation for automated cryo-EM workflows in modern experimental settings.

1 INTRODUCTION

Cryo-electron microscopy (cryo-EM) is a central tool for determining the 3D structures of biological molecules at near-atomic resolution. In a typical single particle cryo-EM experiment, an aqueous solution of molecules is vitrified, imaged as randomly-oriented 2D projections by a transmission electron microscope, and computationally reconstructed into 3D density volumes. A major computational challenge is *ab initio* reconstruction, i.e., recovering 3D structure(s) without any image poses or initial volumes. Over the past decade, methods and software for *ab initio* 3D reconstruction and refinement have matured, enabling routine structure determination for well-behaved, structurally homogeneous (i.e. rigid) biomolecules (Scheres, 2012; Punjani et al., 2017; Lyumkis et al., 2013; Grant et al., 2018). Even for more dynamic biomolecular complexes such as macromolecular machines, standard pipelines are often sufficient to align particle images into a common reference frame, thus enabling downstream advanced heterogeneity analysis tools to recover rich conformational ensembles (Zhong et al., 2021a; Punjani & Fleet, 2021; Nakane et al., 2018; Donnat et al., 2022).

Modern cryo-EM experiments, however, increasingly image less-purified samples containing a mixture of distinct molecular species (Nogales & Mahamid, 2024; Ho et al., 2020). When multiple species are present, there is no common reference frame, and image alignment can no longer be decoupled from resolving the heterogeneity. Poses, per-particle class assignments, and multiple volumes must instead be inferred jointly, a substantially harder optimization. Multiclass *ab initio* reconstruction is the standard approach to this problem, partitioning particle images into a fixed number of classes and reconstructing one volume per class (Punjani et al., 2017). Recent advances, including neural field approaches such as cryoDRGN-AI (Levy et al., 2025) and neural mixture models such as Hydra (Levy et al., 2024), have proposed more expressive models of heterogeneity that are jointly optimized with image poses for heterogeneous *ab initio* reconstruction. Despite this progress, current methods fail when the number of distinct species grows large, as this poses a much more challenging optimization problem (Jeon et al., 2024). In practice, practitioners compensate with ad hoc strategies: running reconstructions repeatedly with different random seeds, manually inspecting and filtering the results, and iterating until satisfactory classes emerge. These workflows are labor-intensive, difficult to reproduce, and scale poorly with mixture complexity.

Here, we observe that multiclass *ab initio* reconstruction jobs, despite failing to resolve a complex mixture in full, carry weak classification signal. This is analogous to the classical weak learning setting (Schapire, 1990), where learners that perform only slightly better than chance can be aggregated into a strong classifier. Building on this connection and related work on cluster ensembles (Strehl & Ghosh, 2002; Fred & Jain, 2005), we introduce *reconstruction ensembles*, a meta-algorithm

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

for systematically generating, weighting, and aggregating *ab initio* reconstruction jobs to resolve complex mixtures that no single run can handle. Using cryoSPARC's multiclass reconstruction methods (Punjani et al., 2017) as weak classifiers, we demonstrate state-of-the-art *ab initio* reconstruction of synthetic and real datasets, including successful recovery of complex mixtures from datasets containing more than tens of species for the first time. The approach is tool-agnostic, requires no modifications to existing reconstruction software, and lays the groundwork for reproducible, automated cryo-EM workflows. In summary, we make the following contributions:

- A meta-algorithm for spawning and aggregating *ab initio* reconstruction algorithms to produce final particle classification assignments,
- State-of-the-art *ab initio* reconstruction of datasets containing complex mixtures, achieving 97% classification accuracy of a 45-class subset of Tomotwin-100 and 75% accuracy on the full Tomotwin-100 dataset (Jeon et al., 2024),
- A workflow automation framework built on the cryoSPARC-tools API that enables reproducible, programmatic control of cryo-EM reconstruction pipelines.

2 Background and related work

2.1 Single particle cryo-EM reconstruction

In single particle cryo-EM, an aqueous solution of molecules is vitrified and imaged in a transmission electron microscope, producing 10^4–10^7 noisy 2D projection images. Each image X_i captures a separate copy of the molecule at an unknown orientation, and can be modeled as (Levy et al., 2024; Vulović et al., 2013):

$$X_i = C_i \cdot \mathcal{P}_{\phi_i} V_i + \eta_i \tag{1}$$

where $V_i : \mathbb{R}^3 \to \mathbb{R}$ is the 3D electron scattering potential (density map), $\phi_i = (R_i, \mathbf{t}_i)$ is the unknown pose consisting of a rotation $R_i \in SO(3)$ and in-plane translation $\mathbf{t}_i \in \mathbb{R}^2$, $\mathcal{P}_{\phi_i}$ is the projection operator that integrates V_i along the imaging axis at orientation ϕ_i, C_i is the contrast transfer function (CTF) of the microscope, and $\eta_i \sim \mathcal{N}(0, \sigma^2)$ is additive noise. The images are recorded on a discrete $D \times D$ grid with typical signal-to-noise ratios between 10^{-1} and 10^{-2}.

The central computational task is to recover V from these projections, which requires jointly solving for the unknown volume and the per-image poses. Expectation-maximization and coordinate ascent algorithms are typically used to find a maximum *a posteriori* estimate of V, marginalizing over the pose posterior for each image (Scheres, 2012). This iterative refinement is sensitive to the initial volume estimate; stochastic gradient descent is commonly used for *ab initio* reconstruction to provide a starting model from scratch (Punjani et al., 2017).

2.2 Multiclass *ab initio* reconstruction

Real cryo-EM samples often contain mixtures of distinct molecular species or conformational states. The standard approach to this *heterogeneity problem* extends the image formation model to assume that images are generated from K independent volumes $V_1, \ldots, V_K$. Inference then requires marginalization over both the per-image poses and class assignment probabilities π_j:

$$\arg \max_{V_1,\ldots,V_K} \sum_{i=1}^{N} \log \sum_{j=1}^{K} \left[\pi_j \int p(X_i \mid \phi, V_j)\, p(\phi)\, d\phi \right] + \sum_{j=1}^{K} \log p(V_j) \tag{2}$$

This formulation, commonly called multiclass refinement or 3D classification, is implemented in major software packages such as RELION (Scheres, 2012) and cryoSPARC (Punjani et al., 2017). In the *ab initio* setting, the algorithm must simultaneously infer both the image class assignments and poses from scratch (i.e. random initialization), making the optimization landscape considerably more challenging than the refinement or classification case where initial volume models or fixed poses are provided, respectively.

A key limitation of multiclass *ab initio* reconstruction is that the number of classes K must be specified in advance and is typically kept small (e.g., $K \leq 10$). When the true number of species M exceeds K, the algorithm produces a coarse partition that merges distinct species into shared

classes. When K is set too large, the optimization frequently fails to converge or produces degenerate solutions. In practice, this means that for complex mixtures containing tens of distinct species, no single *ab initio* run can resolve the full composition of the sample.

2.3 Heterogeneity analysis beyond discrete classes

A substantial body of work has been devoted to expanding the types of heterogeneity that cryo-EM methods can capture (Donnat et al., 2022; Jeon et al., 2024). While early methods modeled heterogeneity as a discrete mixture of independent, homogeneous volumes (Scheres, 2012; Punjani et al., 2017), more recent approaches have moved toward continuous representations of structural variability, including linear subspaces (Punjani & Fleet, 2021), neural fields (Zhong et al., 2021a;b), Gaussian mixture models (Chen & Ludtke, 2021), and parametric deformation fields (Punjani & Fleet, 2023). However, the majority of these methods operate with fixed poses obtained from an upstream consensus reconstruction, limiting their applicability when the sample contains strong compositional heterogeneity that prevents reliable pose estimation in the first place.

In the *ab initio* setting where poses must be inferred jointly with the heterogeneity, fewer methods are available. The multiclass *ab initio* capabilities in RELION and cryoSPARC (Eq. 2) model discrete mixtures but are limited to small K. CryoDRGN-AI (Levy et al., 2025) demonstrated a fully autodecoding framework for *ab initio* heterogeneous reconstruction, and Hydra (Levy et al., 2024) extended this with a mixture of K neural fields to jointly capture compositional and conformational heterogeneity. Stepping beyond the capabilities of a single call to these methods, Lauzirika et al. (2025) proposed a statistical framework for exploring the number of discrete classes using assignment consistencies across replicate runs, an intuition closely related to ours, which we extend through a full ensemble aggregation framework. Despite this progress, resolving complex mixtures containing tens or more distinct species at the *ab initio* stage remains an open challenge, with no existing methods, to our knowledge, able to recover any of the underlying structures from the Tomotwin-100 benchmark dataset (Jeon et al., 2024).

2.4 Workflow automation in cryo-EM

The cryo-EM data processing pipeline has historically been driven by expert users interacting with graphical interfaces. Recently, several efforts have begun to automate portions of this workflow. The cryoSPARC team demonstrated an end-to-end Workflow feature (CryoSPARC et al., 2025), and the CryoWizard pipeline (Yan et al., 2026) automates particle curation using the Cryo-IEF model. CryoSift (Schäfer et al., 2025) provides another example of programmatic interaction with the cryoSPARC suite. Our ensemble framework builds directly on this trend, using the cryoSPARC-tools API to orchestrate the generation and aggregation of reconstruction jobs without manual intervention.

2.5 Weak learning and cluster ensembles

A central result in machine learning is that weak learners, i.e. classifiers performing only slightly better than random chance, can be aggregated into strong learners with substantially better performance guarantees (Schapire, 1990). This observation has driven decades of research on boosting and ensemble methods, producing canonical algorithms such as AdaBoost (Freund & Schapire, 1997) and XGBoost (Chen & Guestrin, 2016).

In the unsupervised setting, where labeled feedback is unavailable, a related subfield of *cluster ensembles* has emerged (Strehl & Ghosh, 2002). Given multiple clustering assignments of the same dataset, the goal is to aggregate them into a single, stronger partition. A canonical approach due to Fred & Jain (2005) constructs a co-assignment matrix recording how often pairs of data points are placed in the same cluster across ensemble members, then applies hierarchical agglomerative clustering to this accumulated evidence. Alternative formulations include graph partitioning (Fern & Brodley, 2004) and information-theoretic objectives (Strehl & Ghosh, 2002). Our weighted similarity score builds directly on the co-assignment matrix framework of Fred & Jain (2005), extending it with coverage-aware masking and interpretable weight heuristics tailored to the structure of cryo-EM reconstruction outputs.

3 METHODS

Our algorithm proceeds in three stages: ensemble generation, weighted similarity scoring, and clustering. We first formalize the tool abstraction and the two primitives for building ensembles (Fig. 1), then describe the aggregation framework that produces the final particle assignments (Fig. 2).

3.1 NOTATION AND TOOL ABSTRACTION

We assume access to a stochastic multiclass *ab initio* reconstruction tool that takes a set of N particle images and assigns each particle to one of K classes. Let $I = \{0, \ldots, N-1\}$ denote the particle index set.

We index the jobs in the ensemble with a single flat counter: $J_0, J_1, \ldots, J_{T-1}$ for T total jobs. Each job J_m is characterized by its input particle subset $I_m \subseteq I$, its class count K_m, and its random seed ρ_m. Annotating the set of possible class labels for a particle to be $[K_m] = \{0, 1, \ldots, K_m - 1\}$, the tool returns a label vector l_m over the input particles:

$$J_m(I_m; K_m, \rho_m) = \texttt{AbInitio}(I_m; K_m, \rho_m) = l_m \in [K_m]^{|I_m|} \tag{3}$$

The label vector partitions I_m into disjoint subsets $I_m^k = \{i \in I_m \mid l_m(i) = k\}$ for $k \in [K_m]$, which we call the **children** of J_m. If the tool produces soft class probabilities rather than hard assignments, we take the argmax as the label. Any other tool-specific parameter is held fixed across the ensemble.

The particles may originate from $M \geq 1$ true underlying species. Our only assumption is that the tool assigns particles better than uniform random, regardless of the relationship between K_m and M. This is sufficient for the tool to serve as a weak classifier.

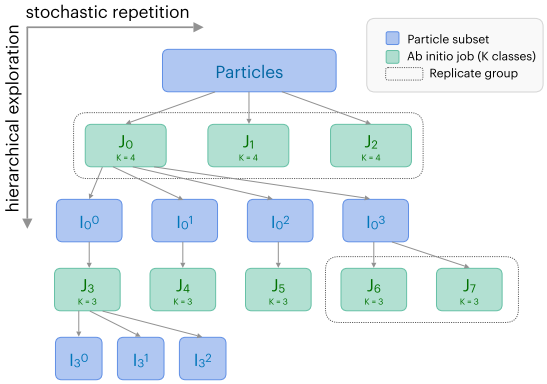


Figure 1: Overview of *ab initio* ensemble generation. Along the horizontal axis, **stochastic repetition** produces replicate jobs that share the same input subset I_m and class count K_m but differ in random seed ρ_m. Along the vertical axis, **hierarchical exploration** launches new jobs on the children I_m^k of a parent job.

3.1.1 ENSEMBLE GENERATION

The ensemble $\mathcal{J} = \{J_0, \ldots, J_{T-1}\}$ is built from two primitives, visually summarized in Figure 1.

Stochastic repetition. Given a particle subset I_m and a class count K_m, this primitive launches R replicate jobs with identical parameters while varying the random seed ρ_m. Cross-replicate consistency in assignments serves as a proxy for classification confidence.

Hierarchical exploration. Given a completed job J_m, this primitive launches a new job on each of its children $I_m^0, \ldots, I_m^{K_m-1}$, with a possibly different class count K'. This divide-and-conquer strategy enables the ensemble to probe a number of true classes that grows exponentially in the tree depth.

These two primitives can be composed freely. The similarity score aggregation (described below) accommodates any job tree or forest, so no particular generation strategy is required.

3.1.2 WEIGHTED SIMILARITY SCORE

Given the completed ensemble $\mathcal{J} = \{J_0, \ldots, J_{T-1}\}$, we aggregate per-job class assignments into a single inter-particle similarity matrix S (Fig. 2). This proceeds in two stages: computing per-job co-assignment matrices, then taking their weighted average.

For each job J_m with input subset $I_m \subseteq I$, the **co-assignment matrix** $C^m \in \{0, 1\}^{N \times N}$ records whether two particles (i, j) received the same class label, i.e. $C_{ij}^m = \mathbb{I}\{l_m(i) = l_m(j)\}$. Furthermore,

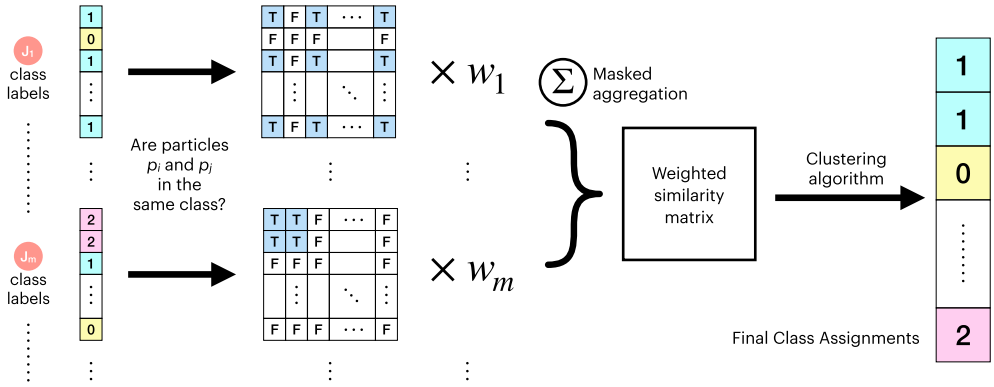


Figure 2: Overview of the job aggregation framework that produces final particle assignments. For every job J_m in the ensemble $\mathcal{J}$, the particle co-assignment matrix is calculated, then averaged with weights w_m to yield a similarity score matrix. The similarity score matrix is fed to any off-the-shelf clustering algorithm that accepts pairwise distance inputs to yield final class assignments.

a **coverage mask** Ω^m tracks which particle pairs were present in the job, masking out any pair that was not present in both, i.e. $\Omega^m_{ij} = \mathbb{I}\{i \in I_m\} \cdot \mathbb{I}\{j \in I_m\}$.

The **similarity** between particles i and j is then the weighted mean of their co-assignment values across all jobs in the ensemble that covered both, i.e. their **coverage-aware weighted average**:

$$S_{ij} \coloneqq \frac{\sum_{m=0}^{T-1} w_m \, \Omega^m_{ij} \, C^m_{ij}}{\sum_{m=0}^{T-1} w_m \, \Omega^m_{ij}} \in [0, 1] \tag{4}$$

The weights in this aggregation routine can be supplied by the expert user if desired. However, instead of requiring the user to manually inspect and assign per-job weights, we also provide **weight heuristics** to automatically compute w_m as a product of three interpretable penalty factors: (1) jobs that underuse their allotted classes (w_m^{eff}), (2) jobs that fragment particles into tiny clusters (w_m^{tiny}), and (3) jobs that funnel most particles into a single dominant class (w_m^{sink}). All three are derived from the class proportions of job J_m. We write $p^k_m = n^k_m / N_m$ for the fraction of the $N_m = |I_m|$ input particles assigned to class k, where $\sum_k n^k_m = N_m$.

$$w_m \coloneqq w_m^{\text{eff}} \cdot w_m^{\text{tiny}} \cdot w_m^{\text{sink}} \tag{5}$$

The **effective-class weight** w_m^{eff} penalizes jobs that concentrate particles into fewer classes than requested. We measure the effective number of occupied classes via the inverse Simpson index (Simpson, 1949) (equivalently, the inverse Herfindahl-Hirschman index (Rhoades, 1993)), K_m^{eff}. The penalty is Gaussian in log-space, so that $w_m^{\text{eff}} = 1$ when classes are perfectly balanced ($K_m^{\text{eff}} = K_m$) and decays smoothly with bandwidth σ as the distribution becomes more skewed:

$$w_m^{\text{eff}} = \exp\left(-\frac{(\log K_m^{\text{eff}} - \log K_m)^2}{2\sigma^2}\right), \quad \text{here} \quad K_m^{\text{eff}} = \frac{1}{\sum_k (p^k_m)^2} \tag{6}$$

The **tiny-class weight** w_m^{tiny} penalizes jobs that produce classes with less than $p^{\min}$ fraction of particles, which are too small for reliable downstream reconstruction. The total probability mass in such classes is $t_m = \sum_k \mathbb{I}\{p^k_m < p^{\min}\} \; p^k_m$, and the penalty decays exponentially on t_m with rate γ. The **sink-class weight** w_m^{sink} penalizes jobs in which one dominant class absorbs most particles, since such a job provides little discriminative signal to the ensemble. Writing $p_m^{\max} = \max_k p^k_m$, the penalty decays exponentially on $p_m^{\max}$ with rate β.

$$w_m^{\text{tiny}} = \exp\left(-\gamma \, t_m\right), \quad w_m^{\text{sink}} = \exp(-\beta \, p_m^{\max}) \tag{7}$$

Alternative functional forms (e.g., hard thresholds) are compatible with the framework, where custom weight heuristics could be engineered at will. For our purposes, the exponential form sufficed while providing simplicity in interpretation.

3.1.3 CLUSTERING

Once S is computed, any clustering algorithm that accepts pairwise similarities (or distances via $1-S$) can be applied. We use agglomerative clustering with average linkage, as it allows straightforward exploration of how assignments change with the number of clusters.

A practical issue arises from the dense structure of S: with average linkage, large groups of weakly similar particles can dominate the merge criterion over smaller groups of strongly similar particles, causing the algorithm to absorb rare species into majority clusters prematurely. To address this, we apply a **sparsification** step before clustering. For each row of S, we retain only the top n_{nbr} strongest similarity scores and zero out the rest, effectively constructing a nearest-neighbor graph. We also zero out the diagonal (trivial self-similarity) and symmetrize the result by taking $S_{ij} \leftarrow \max(S_{ij}, S_{ji})$. The final cluster assignments are obtained by cutting the agglomerative dendrogram at the desired number of classes.

4 RESULTS

We first characterize the performance of the individual *ab initio* reconstruction jobs on datasets of increasing heterogeneity. We then evaluate our ensembling approach on *ab initio* reconstruction of synthetic datasets containing up to 100 distinct biomolecular complexes as well as an experimental cryo-EM dataset. Finally, we analyze how performance scales as a function of ensemble growth and perform hyperparameter ablations of the aggregation step.

4.1 THE *ab initio* RECONSTRUCTION TOOL

Although our approach is tool-agnostic for any multiclass reconstruction algorithm, we use the "Ab-Initio Reconstruction" job in cryoSPARC software suite (Punjani et al., 2017), which has programmatic job management through a Python API exposed in the cryosparc-tools library.

We use only the *default* parameters in version 4.7.1, except the number of classes K and the random seed ρ. Given the extensive tunability of this job, we note that our results could likely be further improved by optimizing the parameters, e.g. the Fourier radius step in smaller molecules (Kim et al., 2025). Here, we focus on highlighting the capability of ensembling even with default job settings.

4.1.1 DATASETS

The Tomotwin-100 dataset, proposed by the CryoBench benchmark suite (Jeon et al., 2024), consists of 100 structures of various molecular weights and symmetries, with 1,000 images each $(D = 128)$, uniformly distributed across the viewing sphere. We focus our analysis on this dataset, since to the best of our knowledge, the complete *ab initio* reconstruction of this dataset has not been reported yet, where the current state-of-the-art analysis provided in Hydra (Levy et al., 2024) only focused on a 3-class subset of this 100-class dataset.

We also analyze a 45-class subset of the asymmetric complexes in Tomotwin-100, which we call Tomotwin-45-asym. Evaluating pose accuracy on the full dataset is complicated by symmetry: when the underlying structure possesses rotational symmetry, an image can be explained by multiple equivalent poses, making pose error ill-defined (Levy et al., 2024; Hodaň et al., 2016). Instead of using non-cryo-EM symmetry-aware pose error functions (Hodaň et al., 2016), we restrict our pose evaluation to structures with C1 symmetry in Tomotwin-100, yielding 45 classes and 45,000 particles.

To test our method beyond synthetic data, we use EMPIAR-10076 (Davis et al., 2016), a standard benchmark for heterogeneous reconstruction of compositional heterogeneity (Davis et al., 2016; Zhong et al., 2021a) consisting of *E. coli* large ribosomal subunit assembly intermediates ($N = 131,899$ images, $D = 320$). The dataset contains four major structural states, two rare species, and a substantial junk fraction. We process the full, unfiltered stack in our reconstruction.

To compute classification accuracy, in order to assess the correct permutation of labels, we follow a simple majority vote within every cluster to determine which ground truth label it maps to. In pose accuracy measurements, the final class assignments are used to generate per-class homogeneous reconstructions, which are then aligned with the mapped ground truth poses with a global rigid-body alignment.

4.2 PERFORMANCE OF THE ONE-SHOT AB INITIO JOBS

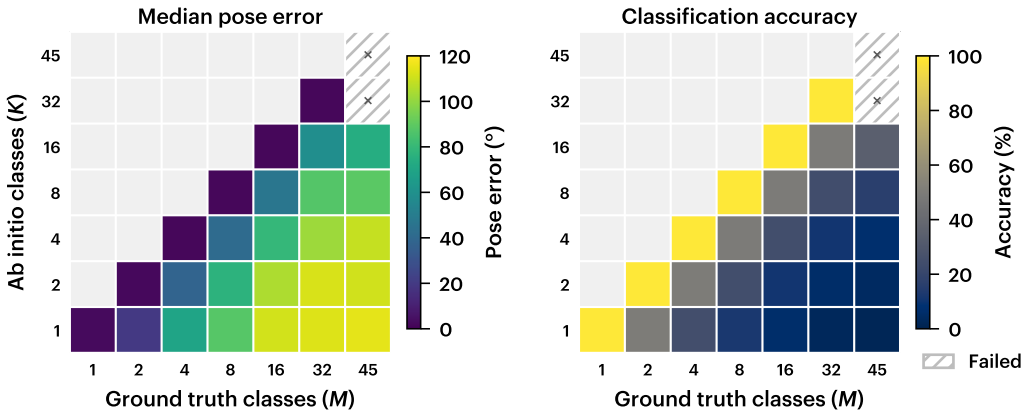


Figure 3: Median angular error (left) and classification accuracy (right) of single-shot ab initio jobs on subsets of Tomotwin-45-asym with increasing heterogeneity.

We first run the cryoSPARC (Punjani et al., 2017) "Ab-Initio Reconstruction" job on subsets of Tomotwin-45-asym with increasing number of ground truth classes ($M = 1, 2, \ldots, 45$) and varying the class number parameter K. This sweep serves two purposes: it establishes a baseline for single-shot reconstruction performance, and it validates the coarse clustering assumption ($K < M$) that our ensemble design relies on. The median angular pose error and classification accuracy are reported in Figure 3, with the complete suite of metrics in Appendix A.

On the diagonal ($K = M$), the job achieves near-perfect classification and pose estimation up to $M = 32$, with median pose errors of 2–3 degrees. As the ground truth complexity grows relative to the job's class budget (moving rightward or downward from the diagonal) both accuracy and pose quality degrade gracefully. On the full dataset ($M = 45$), the job fails to converge at large K values ($K \geq 32$) but still converges for coarser settings ($K \leq 16$), confirming that it can serve as a weak classifier even when it cannot resolve the full mixture.

4.3 PERFORMANCE OF THE AB INITIO ENSEMBLE

Table 1: Heterogeneous ab initio reconstruction performance measured by classification and pose metrics for the Tomotwin benchmark datasets. For the baseline one-shot strategy, the best performing jobs are shown, i.e. $K = 16$ for both datasets.

DATASET	STRATEGY	CLASSIFICATION (%)			POSE ERROR (deg)	
		ACCURACY	PRECISION	MACRO F1	MEAN	MEDIAN
TT-45-ASYM	CRYODRGN2	13.1	10.1	10.9	124.6	128.3
	CRYODRGN-AI	9.3	6.8	7.4	124.2	129.3
	HYDRA ($K = 4$)	4.4	0.4	0.7	124.4	129.4
	ONE-SHOT	34.5	18.8	22.6	78.3	74.1
	ENSEMBLE	**97.4**	**97.4**	**97.4**	**6.11**	**1.71**
TT-100	ONE-SHOT	15.2	3.5	5.5	-	-
	ENSEMBLE	**75.1**	**71.3**	**71.9**	-	-

We evaluate our ensemble method on two datasets of increasing difficulty: the 45-class Tomotwin-45-asym and the full 100-class Tomotwin-100. For both datasets, the ensemble was generated using $K = \{4, 8, 16\}$ job trees at differing $2 - 3$ levels of hierarchical depth with $3 - 5$ replicates per level (Appendix B). For Tomotwin-45-asym, its ensemble totaled 257 jobs and 331.2 GPU-hours with a critical path of 4.3 wall-hours; while for Tomotwin-100, its ensemble totaled 234 jobs and 318.4 hours with a critical path of 4.3 wall-hours. In both cases, default parameters were used for all jobs. Further experimental details are provided in Appendix B.

We compare against four baselines: the best single-shot cryoSPARC run ($K = 16$ for both datasets), and three heterogeneous reconstruction methods: cryoDRGN2 (Zhong et al., 2021b), cryoDRGN-AI (Levy et al., 2025), and Hydra ($K = 4$) (Levy et al., 2024). These three methods were run with default settings (Appendix B). Because these methods were designed for smaller mixtures, we include them primarily to illustrate the difficulty of the 45-class setting rather than as matched comparisons. Results are summarized in Table 1.

On Tomotwin-45-asym, the ensemble achieves 97.4% classification accuracy and a median pose error of 1.71°, compared to 34.5% accuracy and 74.1° median pose error for the best single-shot run. The three neural-field baselines produce pose errors near the uniform-random baseline of ∼130°, indicating that these methods cannot resolve the mixture at this complexity. The ensemble method fully reconstructs this 45-class mixture, as seen from the volume renderings in Appendix C. Furthermore, on Tomotwin-100, the ensemble reaches 75.1% accuracy – five times the best one-shot job's 15.2%, which its $K = 16$ caps at 16%. To our knowledge, neither the 45-class nor the 100-class Tomotwin benchmarks have been previously resolved at the *ab initio* stage; the prior state of the art, Hydra (Levy et al., 2024), analyzed a 3-class subset. Our results therefore represent an order-of-magnitude expansion in the mixture complexity that *ab initio* methods can handle.

4.4 Scaling analysis in ensemble growth

To understand how classification accuracy scales with compute, we systematically vary both the depth of the job tree (hierarchical exploration) and the number of random replicates (stochastic repetition), sweeping over $K \in \{4, 8, 16\}$. To simplify the configuration space, each ensemble uses a single K value across all jobs; the reported accuracies are therefore lower bounds on what mixed-K ensembles could achieve. Figure 4 plots the resulting accuracy-compute trade-off surfaces for Tomotwin-45-asym and Tomotwin-100, with further details in Appendix D.

On both datasets, accuracy typically improves with the first few levels of hierarchical depth and with additional replicates. Because replicates within a level run in parallel, the wall-clock time scales with tree depth rather than total job count, growing only logarithmically in the ensemble size. A lone tree, the recursive workflow a practitioner runs by hand, stays below 55% (Appendix D).

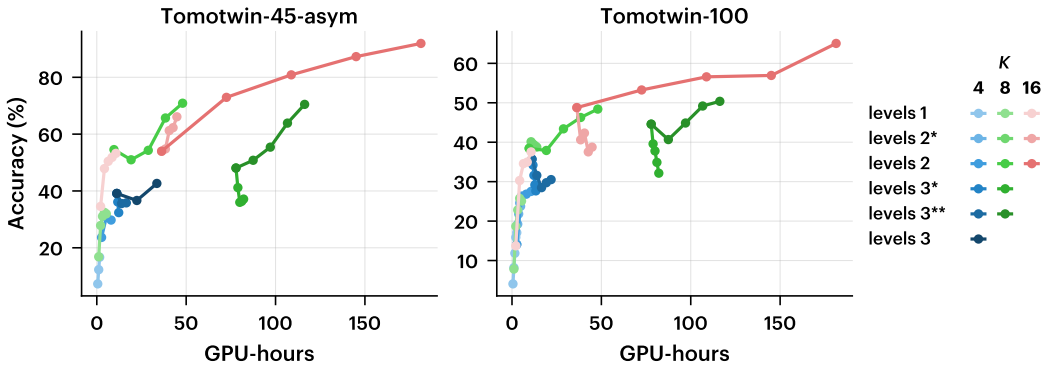


Figure 4: Utility-cost frontiers of scaling job ensembling for Tomotwin-45-asym (left) and Tomotwin-100 (right). The vertical (utility) axis is classification accuracy, and the horizontal (cost) axis is the number of GPU-hours used. Each curve shows ensembling up to 5 replicates and different curves indicate different hierarchical levels with a fixed class K.

4.5 Real dataset

We apply our ensemble method to EMPIAR-10076 (Davis et al., 2016), processing the full unfiltered particle stack ($N = 131{,}899$) without removing junk particles. The ensemble was generated using 3 replicates of $\{4, 8, 16\}$-class jobs at 2 levels of hierarchical depth, and 6-class jobs at 3 levels of hierarchical depth, totaling 222 jobs and 248.8 GPU-hours with a critical path of 4.3 wall-hours. We cut the final dendrogram at 7 classes.

The ensemble successfully separates all four major LSU assembly intermediates (classes B–E) from the junk fraction without manual filtering. Per-class homogeneous refinement yields reconstructions at 3.4–4.0 $\mathring{A}$ resolution for all four states (further experimental details, maps, and FSC curves are in Appendix E). Figure 5 compares the particle counts in our predicted classes against the published annotations (Davis et al., 2016), as well as single-shot *ab initio* baselines. Unlike the single-shot baselines which miss out on the B and C major classes, our ensemble class populations are in broad agreement with the published labels, with the ensemble additionally resolving sub-states within classes D and E that are consistent with the finer partitions reported in prior work (Zhong et al., 2021a; Davis et al., 2016).

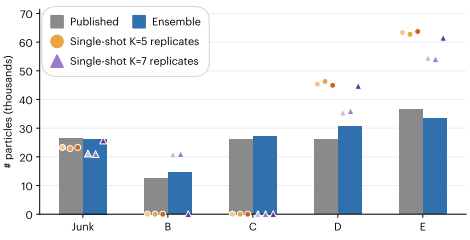


Figure 5: Particle counts of our reconstruction ensembling approach, the published major labels (Davis et al., 2016), and single-job baselines (each repeated with 3 different seeds).

4.6 Hyperparameter ablations

The aggregation step has five hyperparameters: three exponential rates in the weight heuristics (σ, β, γ), a minimum class size threshold $p^{\min}$, and the sparsification neighborhood size n_{nbr}. We sweep all five on Tomotwin-45-asym (Appendix F) and find that the weight parameters have minimal impact (fewer than 10 percentage points of variation), while n_{nbr} is the most consequential choice, producing swings of up to 43.4 percentage points. The main results in Table 1 use $(\sigma, \beta, \gamma, p^{\min}, n_{\text{nbr}}) = (0.6,\ 6,\ 10,\ 1/90,\ 100)$.

5 Discussion

We demonstrate that ensembling *ab initio* reconstruction jobs can resolve complex mixtures in cryo-EM beyond the reach of any single run or any existing method. Because constituent jobs used default settings, we believe these results represent a lower bound on what the framework can achieve with task-specific tuning. By reducing the problem of resolving complex mixtures to the orchestration of fast, coarse reconstruction jobs, the approach opens a path toward high-throughput heterogeneous reconstruction in modern experimental settings where samples contain tens or more distinct species. In doing so, it replaces the ad hoc classification schemes practitioners currently rely on with a systematic procedure that scales with compute rather than manual labor.

The aggregation framework is deliberately flexible. In our experiments, we considered ensembles with a fixed K and a predefined replication strategy for scaling analysis, but the aggregation accepts any collection of multi-class reconstruction jobs regardless of their parameters, input subsets, or tree structure. A practitioner could mix jobs with different K values, incorporate hand-picked runs targeting specific subpopulations, or feed in results from entirely different software packages. This same flexibility makes the framework amenable to autonomous or agent-driven exploration strategies that adaptively allocate compute based on intermediate ensemble results.

Several limitations warrant discussion. First, although the aggregation function is principled, it introduces hyperparameters that encode priors on cluster sizes and uniformity. Our ablations show robustness to the weight heuristic parameters, but sensitivity to the sparsification neighborhood size n_{nbr}. Whether our default value for n_{nbr} generalizes across diverse real datasets remains to be tested. Second, our method classifies particles but does not explicitly model junk; on EMPIAR-10076, junk particles were separated into their own cluster, but our meta-algorithm could benefit from a dedicated filtering step or treatment of a junk or outlier class. Third, the $N \times N$ similarity matrix becomes a memory bottleneck for large particle stacks.[1] Sparse storage and graph-based clustering algorithms that avoid materializing the full matrix are natural remedies to enable scaling to datasets with millions of particles. Addressing these limitations would bring the framework closer to routine deployment on the increasingly complex mixtures encountered in modern cryo-EM experiments, including cell lysates, crude extracts, and *in situ* imaging.

[1]The required memory was in the tens of GBs for a dataset with $N = 131{,}899$ particles.

AI USE STATEMENT

In this work, we used generative AI tools for helping develop theoretical models or conceptual frameworks, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, supporting qualitative and thematic data analysis, and interpreting results. We have not used generative AI tools for generating synthetic data sets, cleaning and reformatting datasets, and formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, assisting with translation are not applicable to this work. Additionally, we used generative AI tools for creating or editing software code. We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness by 2 authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Both benchmarks are public: Tomotwin-100 through CryoBench (Jeon et al., 2024) and EMPIAR-10076 (Davis et al., 2016). Every job used cryoSPARC's default *ab initio* parameters apart from K and the seed. The ensembles, their compute and the aggregation hyperparameters are specified in Appendix B and Appendix E, the one-shot sweep in Appendix A, the scaling ensembles and timing model in Appendix D, and the hyperparameter sweep in Appendix F. Code will be made publicly available upon publication, and an anonymized version can be shared with reviewers during the discussion period upon request.

REFERENCES

Phipps Arabie and Scott A Boorman. Multidimensional scaling of measures of distance between partitions. *Journal of Mathematical Psychology*, 10(2):148–203, 1973.

Muyuan Chen and Steven J Ludtke. Deep learning-based mixed-dimensional Gaussian mixture model for characterizing variability in cryo-EM. *Nature Methods*, 18(8):930–936, 2021.

Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 785–794, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939785.

Team CryoSPARC, Kelly Barber, Hannah Bridges, Suhail Dawood, Katherine Elder, Nick Frasser, Fiona Hu, Serena Liu, Michael McLean, Ryan Narine, et al. End-to-end automation of repeat-target cryo-EM structure determination in CryoSPARC. *bioRxiv*, 2025. doi: 10.1101/2025.10.17.682689.

Joseph H Davis, Yong Zi Tan, Bridget Carragher, Clinton S Potter, Dmitry Lyumkis, and James R Williamson. Modular assembly of the bacterial large ribosomal subunit. *Cell*, 167(6):1610–1622, 2016.

Claire Donnat, Axel Levy, Frederic Poitevin, Ellen D Zhong, and Nina Miolane. Deep generative modeling for volume reconstruction in cryo-electron microscopy. *Journal of Structural Biology*, 214(4):107920, 2022.

Xiaoli Zhang Fern and Carla E Brodley. Solving cluster ensemble problems by bipartite graph partitioning. In *Proceedings of the Twenty-First International Conference on Machine Learning*, pp. 36, 2004.

Edward B Fowlkes and Colin L Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78(383):553–569, 1983.

Ana LN Fred and Anil K Jain. Combining multiple clusterings using evidence accumulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6):835–850, 2005.

Yoav Freund and Robert E Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

Timothy Grant, Alexis Rohou, and Nikolaus Grigorieff. *cis*TEM, user-friendly software for single-particle image processing. *eLife*, 7:e35383, 2018.

Chi-Min Ho, Xiaorun Li, Mason Lai, Thomas C Terwilliger, Josh R Beck, James Wohlschlegel, Daniel E Goldberg, Anthony WP Fitzpatrick, and Z Hong Zhou. Bottom-up structural proteomics: cryoEM of protein complexes enriched from the cellular milieu. *Nature Methods*, 17(1):79–85, 2020.

Tomáš Hodaň, Jiří Matas, and Štěpán Obdržálek. On evaluation of 6D object pose estimation. In *European Conference on Computer Vision*, pp. 606–619. Springer, 2016.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.

Minkyu Jeon, Rishwanth Raghu, Miro Astore, Geoffrey Woollard, Ryan Feathers, Alkin Kaz, Sonya Hanson, Pilar Cossio, and Ellen Zhong. CryoBench: Diverse and challenging datasets for the heterogeneity problem in cryo-EM. *Advances in Neural Information Processing Systems*, 37: 89468–89512, 2024.

Kookjoo Kim, Huan Li, and Oliver B. Clarke. High-resolution ab initio reconstruction enables cryo-EM structure determination of small particles. *bioRxiv*, 2025. doi: 10.1101/2025.09.08.674935.

O Lauzirika, M Pernica, D Herreros, E Ramírez-Aportela, J Krieger, M Gragera, M Iceta, P Conesa, Y Fonseca, J Jiménez, et al. How many (distinguishable) classes can we identify in single-particle analysis? *Acta Crystallographica Section D*, 81(10):535–544, Oct 2025. doi: 10.1107/S2059798325007831.

Axel Levy, Rishwanth Raghu, David Shustin, Adele Peng, Huan Li, Oliver Clarke, Gordon Wetzstein, and Ellen Zhong. Mixture of neural fields for heterogeneous reconstruction in cryo-EM. *Advances in Neural Information Processing Systems*, 37:56988–57017, 2024.

Axel Levy, Rishwanth Raghu, Ryan J Feathers, Michal Grzadkowski, Frédéric Poitevin, Jake D Johnston, Francesca Vallese, Oliver Biggs Clarke, Gordon Wetzstein, and Ellen D Zhong. CryoDRGN-AI: neural ab initio reconstruction of challenging cryo-EM and cryo-ET datasets. *Nature Methods*, 22(7):1486–1494, 2025.

Dmitry Lyumkis, Axel F Brilot, Douglas L Theobald, and Nikolaus Grigorieff. Likelihood-based classification of cryo-EM images using FREALIGN. *Journal of Structural Biology*, 183(3): 377–388, 2013.

Takanori Nakane, Dari Kimanius, Erik Lindahl, and Sjors HW Scheres. Characterisation of molecular motions in cryo-EM single-particle data by multi-body refinement in RELION. *eLife*, 7:e36861, 2018.

Eva Nogales and Julia Mahamid. Bridging structural and cell biology with cryo-electron microscopy. *Nature*, 628(8006):47–56, 2024.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Eric F Pettersen, Thomas D Goddard, Conrad C Huang, Elaine C Meng, Gregory S Couch, Tristan I Croll, John H Morris, and Thomas E Ferrin. UCSF ChimeraX: Structure visualization for researchers, educators, and developers. *Protein Science*, 30(1):70–82, 2021.

Ali Punjani and David J Fleet. 3D variability analysis: Resolving continuous flexibility and discrete heterogeneity from single particle cryo-EM. *Journal of Structural Biology*, 213(2):107702, 2021.

Ali Punjani and David J Fleet. 3DFlex: determining structure and motion of flexible proteins from cryo-EM. *Nature Methods*, 20(6):860–870, 2023.

Ali Punjani, John L Rubinstein, David J Fleet, and Marcus A Brubaker. cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination. *Nature Methods*, 14(3):290–296, 2017.

Stephen A Rhoades. The Herfindahl-Hirschman Index. *Federal Reserve Bulletin*, 79:188, 1993.

J-H Schäfer, Austin Calza, Keenan Hom, Puneeth Damodar, Ruizhi Peng, Nebojša Bogdanović, Gabriel C Lander, Scott M Stagg, and Michael A Cianfrocco. CryoSift: an accessible and automated CNN-driven tool for cryo-EM 2D class selection. *Structural Biology and Crystallization Communications*, 81(12):517–526, 2025.

Robert E Schapire. The strength of weak learnability. *Machine Learning*, 5(2):197–227, 1990.

Sjors HW Scheres. RELION: implementation of a Bayesian approach to cryo-EM structure determination. *Journal of Structural Biology*, 180(3):519–530, 2012.

Edward H Simpson. Measurement of Diversity. *Nature*, 163(4148):688–688, 1949.

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3(Dec):583–617, 2002.

Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 1073–1080, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553511.

Miloš Vulović, Raimond BG Ravelli, Lucas J van Vliet, Abraham J Koster, Ivan Lazić, Uwe Lücken, Hans Rullgård, Ozan Öktem, and Bernd Rieger. Image formation modeling in cryo-electron microscopy. *Journal of Structural Biology*, 183(1):19–32, 2013.

Yang Yan, Shiqi Fan, Fajie Yuan, and Huaizong Shen. A comprehensive foundation model for cryo-EM image processing. *Nature Methods*, 23:88–95, 2026.

Ellen D Zhong, Tristan Bepler, Bonnie Berger, and Joseph H Davis. CryoDRGN: reconstruction of heterogeneous cryo-EM structures using neural networks. *Nature Methods*, 18(2):176–185, 2021a.

Ellen D Zhong, Adam Lerer, Joseph H Davis, and Bonnie Berger. CryoDRGN2: Ab initio neural reconstruction of 3D protein structures from real cryo-EM images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4066–4075, 2021b.

A METRIC HEATMAPS OF ONE-SHOT *ab initio* RUNS

In this section, experiments with $K > M$ were also included to complete the single-shot exploration grid.

Figure 6 reports the mean and median pose error metrics. The geodesic error is the rotation angle (in degrees), and the squared error is the squared error between the true and predicted rotation matrices.

Figure 7 reports the classification metrics (accuracy, precision, macro F1). Note that for the calculation of these metrics, we find the most abundant true label in each predicted class to map it to that true label, and in $K > M$ regime, this mapping is not one-to-one, therefore inducing a propensity for higher scores. Conversely, for $K < M$ it caps accuracy at K/M on balanced data: the best one-shot jobs of Table 1 ($K = 16$) score 34.5% against a cap of $16/45 = 35.6\%$ on Tomotwin-45-asym, and 15.2% against $16/100 = 16\%$ on Tomotwin-100.

Figure 8 reports the clustering metrics calculated before the mapping mentioned above, therefore providing an alternative heuristic of the ability of the job. The metrics used are adjusted Rand index (ARI) (Hubert & Arabie, 1985), adjusted mutual information (AMI) (Vinh et al., 2009), normalized mutual information (NMI), Fowlkes–Mallows index (FMI) (Fowlkes & Mallows, 1983), and variation of information (VI) (Arabie & Boorman, 1973). Out of these; ARI, AMI, and NMI are not well defined if one of the label assignments only has a single class, therefore those cells at the boundaries are set to 0. All of these metrics were calculated using scikit-learn library (Pedregosa et al., 2011).

In all figures reported in this section, failed jobs where no convergence was achieved are explicitly annotated. The results overall highlight the graceful degradation of the merit of the job as the true heterogeneity M varies with respect to K (i.e., off-diagonal directions).

In total, 49 jobs were executed on Tomotwin-45-asym, and 64 jobs were executed on Tomotwin-100.

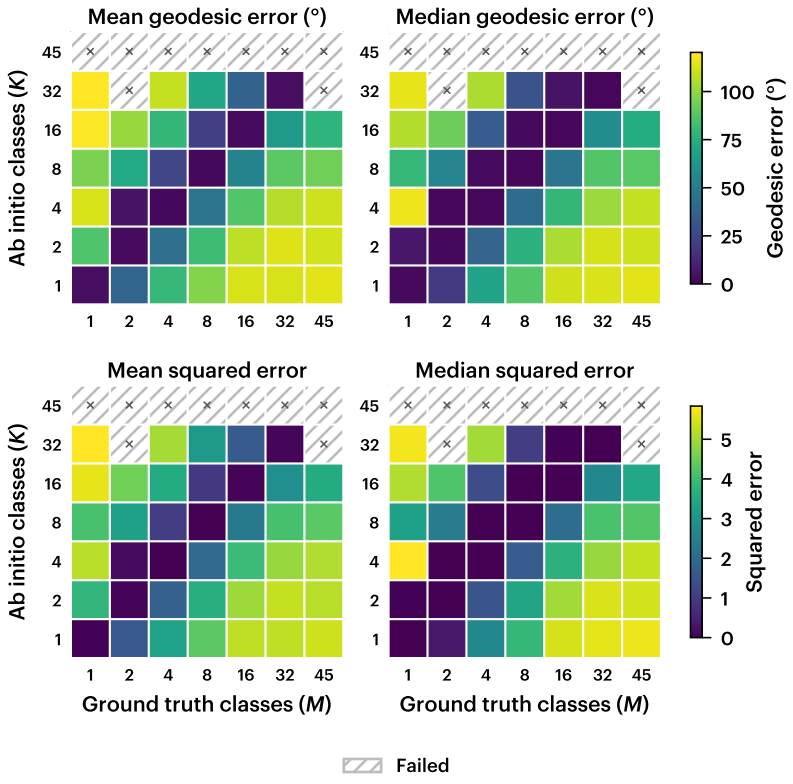


Figure 6: Mean (left) and median (right) pose error metrics of the single-shot *ab initio* runs on subsets of Tomotwin-45-asym with increasing heterogeneity.

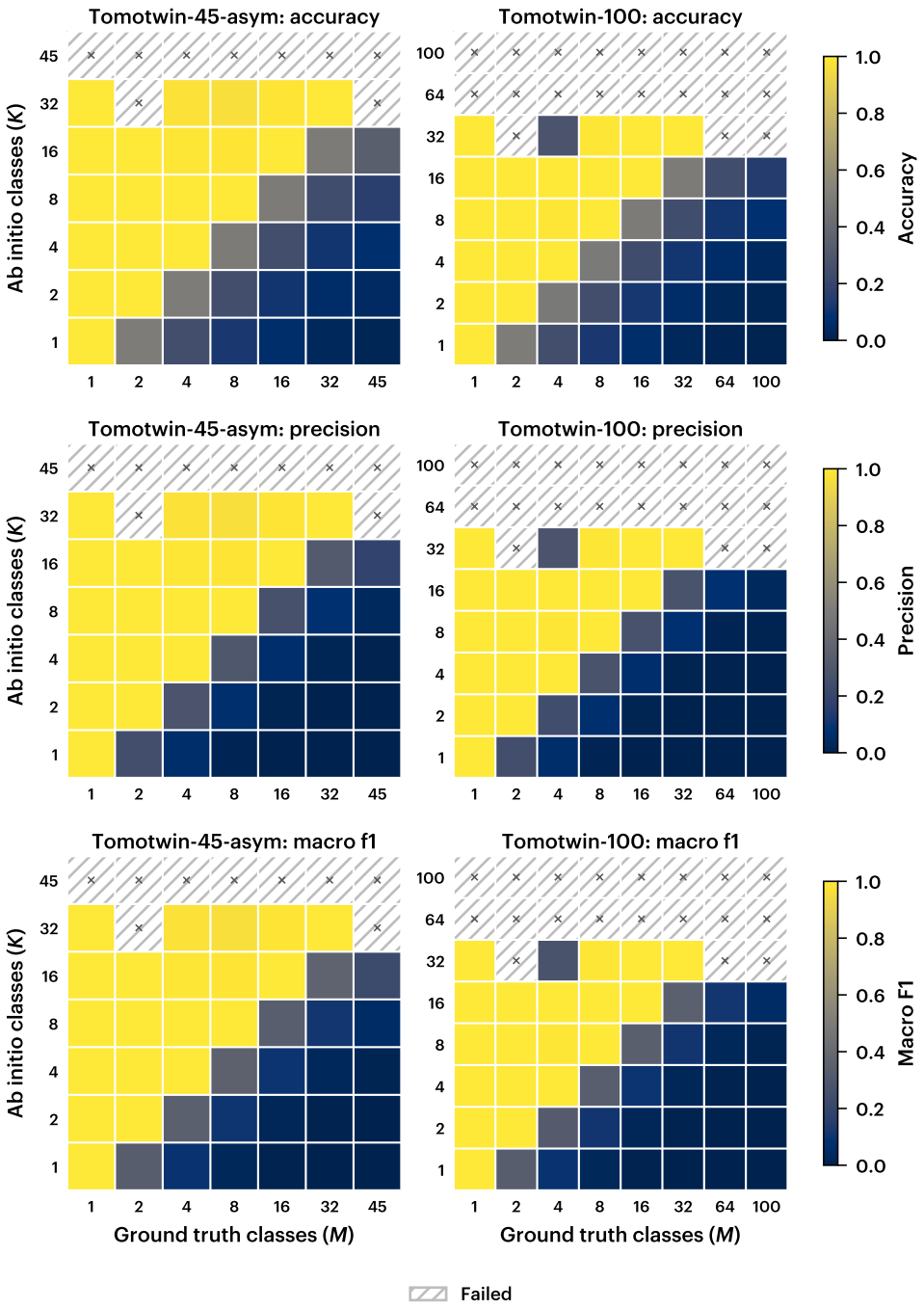


Figure 7: Classification metrics of the single-shot *ab initio* runs on subsets of Tomotwin-45-asym (left) and Tomotwin-100 (right) with increasing heterogeneity.

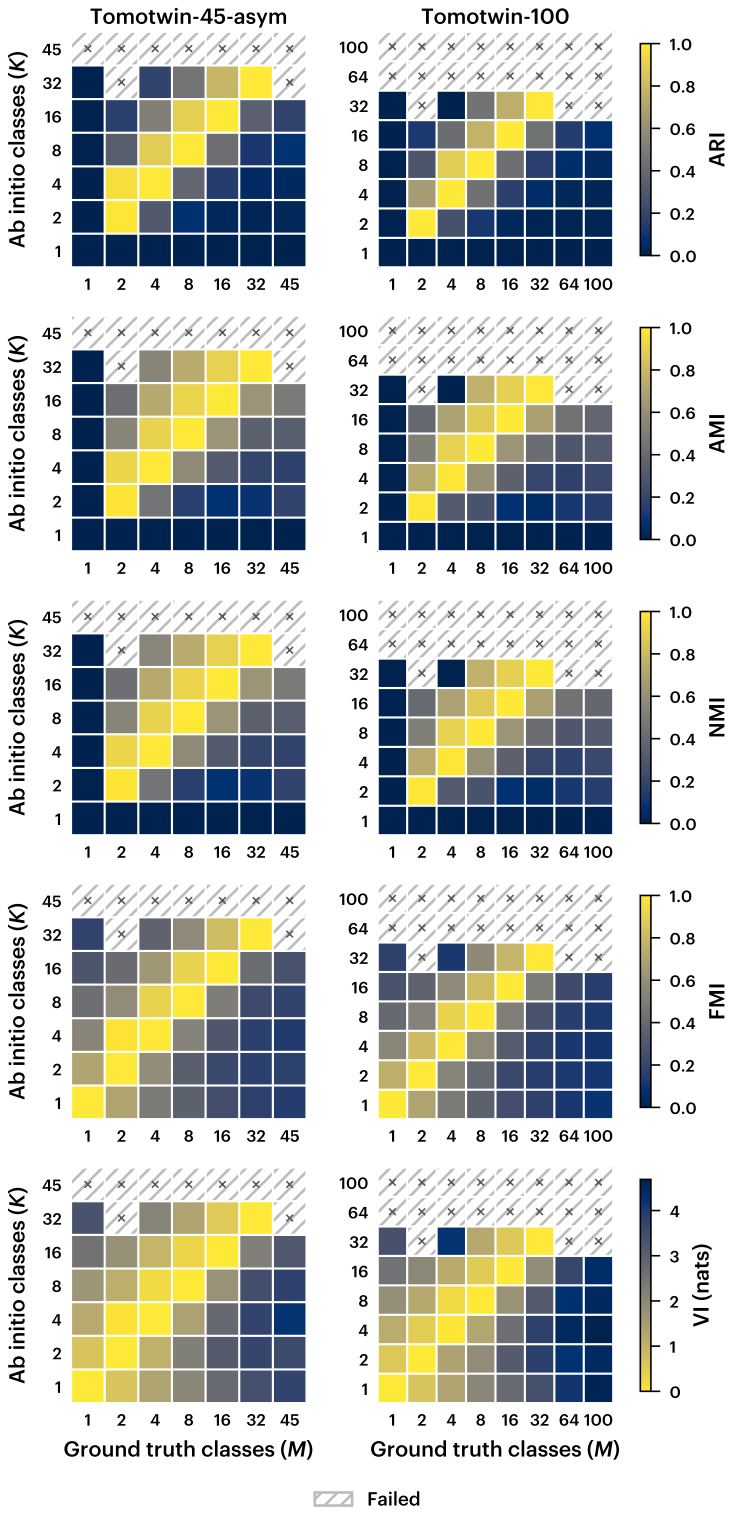


Figure 8: Clustering metrics of the single-shot *ab initio* runs on subsets of Tomotwin-45-asym (left) and Tomotwin-100 (right) with increasing heterogeneity.

B EXPERIMENTAL DETAILS OF THE MAIN RESULTS

B.1 BASELINE METHODS IN TOMOTWIN-45-ASYM

Following baseline method settings were used on Tomotwin-45-asym with classification and pose accuracies as measurement of merit.

- **cryoDRGN2**: The **abinit_het** command in cryoDRGN v3.4 has been used with the canonical latent dimensionality of 8, all other parameters kept at default. The model has been trained for 20 epochs on 1 H100 in 4 hours. Using the analyze command, the resulting latent space is clustered with 45 k-means centroids, yielding the labeling subject to classification accuracy calculation.
- **cryoDRGN-AI**: The **train** command in DRGN-AI repository has been used with the canonical "autodecoder" architecture configuration, all parameters kept at default. The model has been trained for 111 epochs on 1 H100 in 3 hours. Using the **analyze** command, the resulting latent space is clustered with 45 k-means centroids, yielding the labeling subject to classification accuracy calculation.
- **Hydra**: The **train** command in Hydra repository has been used with the default configuration as DRGN-AI. Hydra was not tested on large K values, and its GPU parallelization strategy does not trivially lend itself to a multi-node multi-GPU setup, therefore $K = 45$ run is not viable. Facing this computational bottleneck, we trained a mixture of $K = 4$, for 10 epochs, on 2 H100s, in 6 wall-hours. The resulting 4-class probability distributions of the particles were used to find the most probable class assignment at each particle.

B.2 POSE ERROR

Poses are comparable only within a shared frame. Each predicted class is mapped to its most abundant true class, and one global rotation aligns all of its particles to that class's frame, so particles of other true classes in the same predicted class are scored in the wrong frame. They inflate the mean more than the median, which is why the median is our primary pose metric: the Tomotwin-45-asym ensemble's mean error is 6.11° against a median of 1.71° (Table 1).

B.3 HYPERPARAMETER SELECTION

For the results reported in Table 1, for both Tomotwin-45-asym and Tomotwin-100, the set of best performing hyperparameters in the hyperparameter ablation study (Appendix F) has been used:

$$(\sigma, \beta, \gamma, p^{\min}, n_{\text{nbr}}) = (0.6, 6, 10, 1/90, 100) \tag{8}$$

B.4 JOB ENSEMBLE GENERATION

Recall that we define a K-class, L-level job hierarchy by a tree of K-class jobs, executed on their respective parent's children particle stacks in the tree structure, with a tree of height $L - 1$. Therefore, one such hierarchy consists of $1 + K + \cdots + K^{L-1}$ jobs. The critical path of this tree would then include L jobs (length of sequential dependency), and using the job timing assumption $t(K) = 8K(\text{min})$ (Appendix D), the critical path would take $t_c = 8KL(\text{min})$. Assuming scalable compute access (e.g., cloud instances), the realized wall time could be as low as the critical path.

For Tomotwin-45-asym, our ensemble includes the following jobs, which are also listed on Table 2:

- 3 random replicates each of, 3-level hierarchies, of $K = 4$.
- 1 random replicate of, 3-level hierarchy, of $K = 8$.
- 4 random replicates each of, 2-level hierarchies, of $K = 8$.
- 5 random replicates each of, 2-level hierarchies, of $K = 16$.

Table 2: Job ensemble specification for the Tomotwin-45-asym analysis. The total number of jobs was 257, total GPU-hours used was 331.2, and the longest critical path was 4.3 hours.

# Replicates	K	Levels	Total Job Count	Total GPU-hours	Critical path (wall-hours)
3	4	3	63	33.6	1.6
4	8	2	36	38.4	2.1
1	8	3	73	77.9	3.2
5	16	2	85	181.3	4.3
			257	331.2	4.3

For Tomotwin-100, our ensemble includes the job hierarchies listed on Table 3:

Table 3: Job ensemble specification for the Tomotwin-100 analysis. The total number of jobs was 234, total GPU-hours used was 318.4, and the longest critical path was 4.3 hours.

# Replicates	K	Levels	Total Job Count	Total GPU-hours	Critical path (wall-hours)
4	4	2	20	10.7	1.1
1	4	3	21	11.2	1.6
4	8	2	36	38.4	2.1
1	8	3	72	76.8	3.2
5	16	2	85	181.3	4.3
			234	318.4	4.3

In our cryoSPARC setup, we had access to ≈ 50 parallel GPU job submission lanes, therefore both of these ensembles could be done overnight.

C VOLUME RENDERINGS OF TOMOTWIN-45-ASYM RESULTS

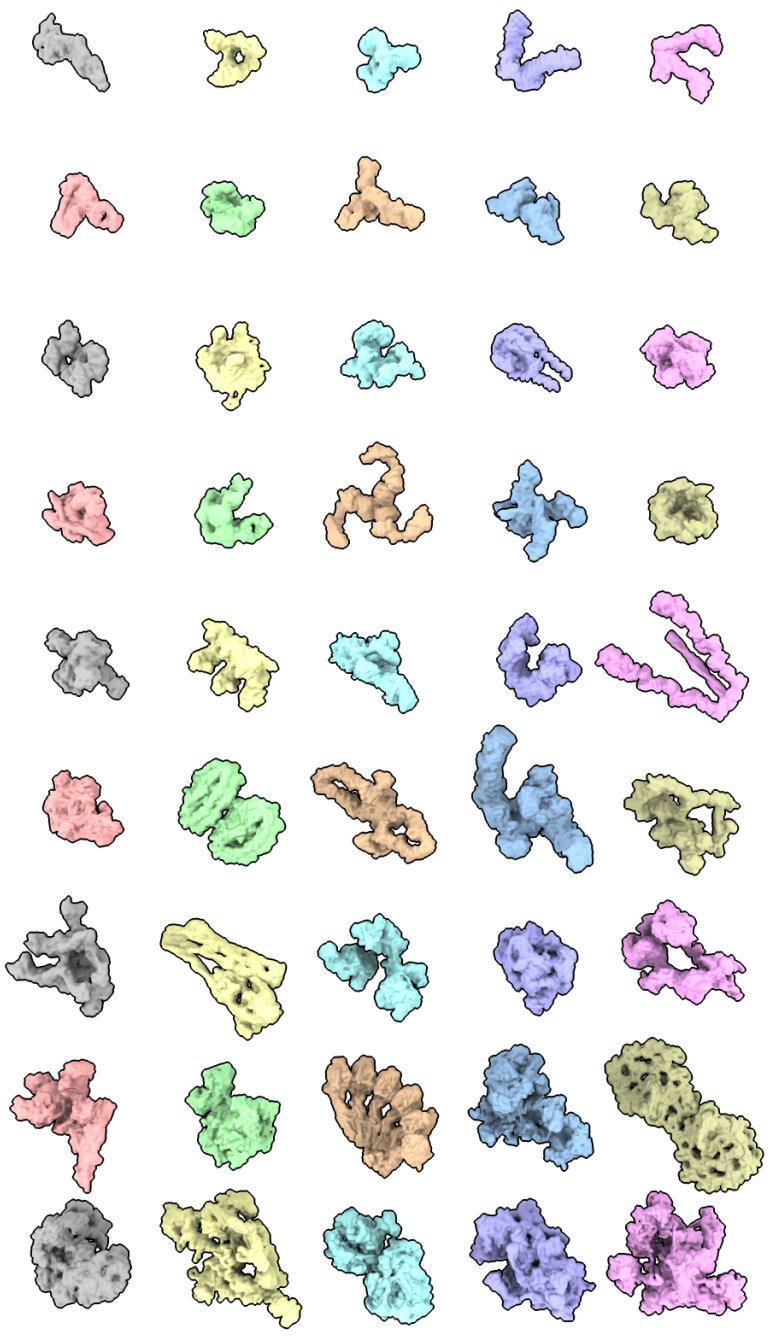


Figure 9: The volumes obtained from Tomotwin-45-asym ensemble classification, as reported in Table 1. The resulting particle classes were fed into homogeneous refinement to yield these refined volumes per each class, and the renderings were done in ChimeraX (Pettersen et al., 2021).

D ENSEMBLE SCALING EXPERIMENTS

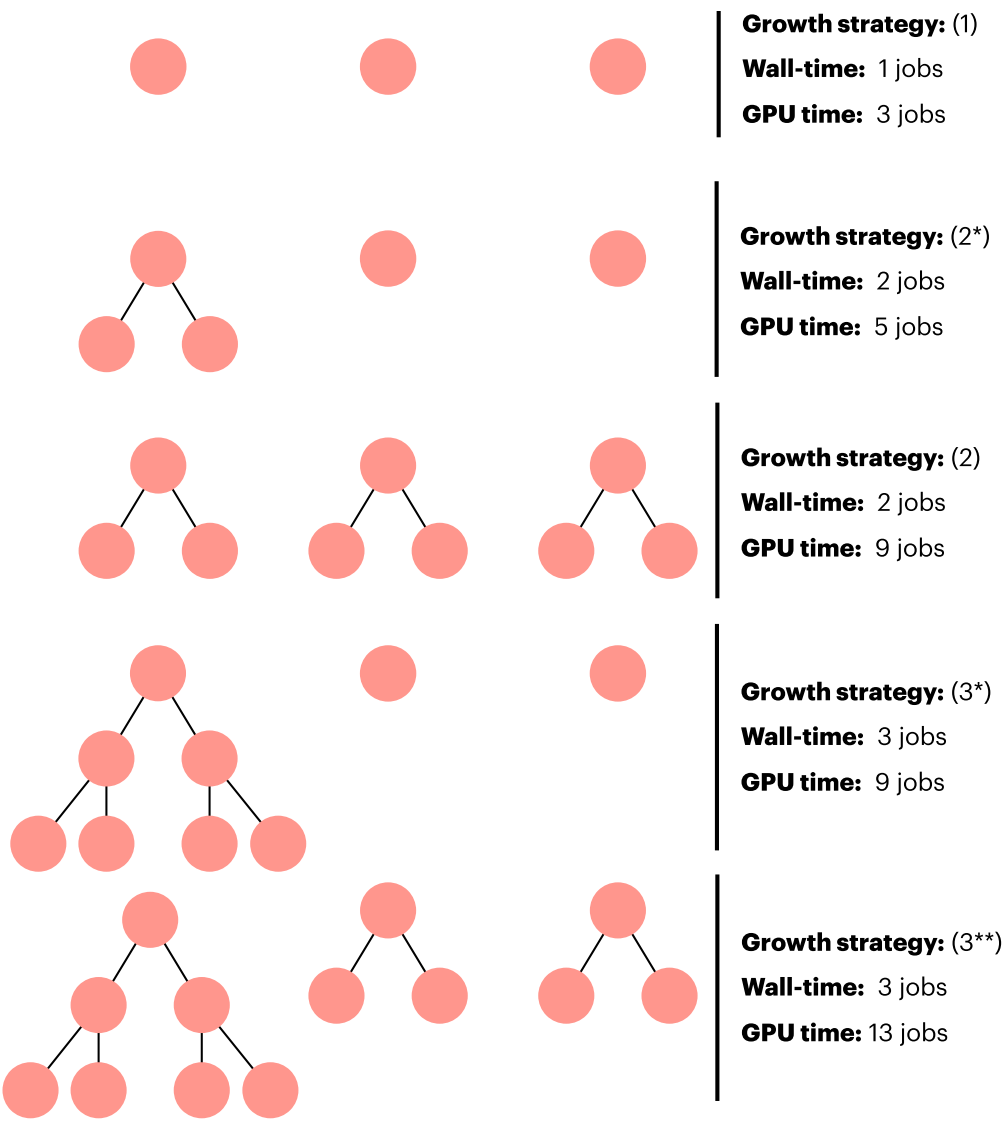


Figure 10: The job ensemble growth strategies employed in the scaling analysis, with the corresponding level annotations and computational costs. All strategies are demonstrated with 3 random replicates. Time is in the unit of jobs, where each job is assumed to require similar time for completion (under a constant K).

In generating the job forests that our method would aggregate as an ensemble, we face the task of enumerating the combinatorial search space of what a job ensemble could consist of. Even in our two-primitive formulation of the growth, and ignoring all the potential parametrizations one can utilize the modern *ab initio* reconstruction tools with, we still face the questions of: (1) what depths of job trees to generate, (2) how many replicates of which depths to generate, (3) with what K values to generate the constituent jobs.

As a reduction of the search space to highlight the first two questions, we therefore kept the K values constant per each ensemble we analyzed in the scaling experiments. Therefore, the K-knob, for the purposes of the enumeration provided here, is elevated from a per-job parameter to a per-ensemble

parameter. After this adjustment, the depth/width questions are highlighted as the principal axes of ensemble growth.

As a further reduction for the purposes of the scaling experiment, we only allow trees at maximum two different heights: the singular *deep* tree, and the *shallow* replicate trees. Therefore, in a given strategy, the *level* number indicates the height+1 of the deep tree, and the number of stars indicate the height+1 of the shallow trees. This naming was chosen to highlight the fact that, assuming abundant spot compute availability for horizontal scaling, the height of the *deep* tree determines the wall-time required for the completion of the job ensemble (since other trees could be executed in parallel). These ensemble growth strategies are demonstrated in Figure 10.

The horizontal dimension of the ensemble generation is the random replicate generation. For the purposes of our enumeration, we include all the shallow trees and the deep tree in the number of random replicates R. For example, a 5-replicate (3**)-strategy ensemble consists of 1 deep tree of height 2, and 4 shallow trees of height 1.

In the scaling trade-off frontiers plotted in Figure 4, each line consists of different random replicates under a constant K and strategy. The base colors (red, green, blue) describe the K value of the ensembles considered, and the hue (light to dark, with increasing GPU-hours) describes the strategy employed.

A single tree ($R = 1$) is the recursive workflow a practitioner runs by hand: classify, then classify each class again. Aggregated with our pipeline, no single tree exceeds 54.6% on Tomotwin-45-asym ($K = 8$, 2 levels, 9 jobs) or 48.8% on Tomotwin-100 ($K = 16$, 2 levels, 17 jobs), and a third level does not help: the $K = 8$, 3-level tree (73 jobs) scores 48.1% and 44.6%. Every ensemble of Figure 4 beyond these adds replicates.

For the time calculation, we observed that the execution time of the jobs scaled linearly with K (8 minutes per K), hence that fixed factor is assumed: $t(K) = 8K(\text{min})$. Each job ran on a single 80 GB GPU, an H100 or an A100, and $t(K)$ is applied to both. The wall-times [i.e., the critical path of sequential dependency, $t_c = 8KL(\text{min})$] of the ensembles are provided in Table 4.

Table 4: Wall-time (in hours) of different ensemble generation strategies.

	Growth Strategy				
K	1	2*	2	3*	3**
4	0.53	1.07	1.07	1.60	1.60
8	1.07	2.13	2.13	3.20	3.20
16	2.13	4.27	4.27	6.40	6.40

Scaling analyses for both Tomotwin-45-asym and Tomotwin-100 have been conducted with the ensemble aggregation hyperparameters kept constant at:

$$(\sigma, \beta, \gamma, p^{\text{min}}, n_{\text{nbr}}) = (0.45, 5, 10, 1/450, 400) \tag{9}$$

E RECONSTRUCTION OF THE BACTERIAL 50S RIBOSOMAL SUBUNIT

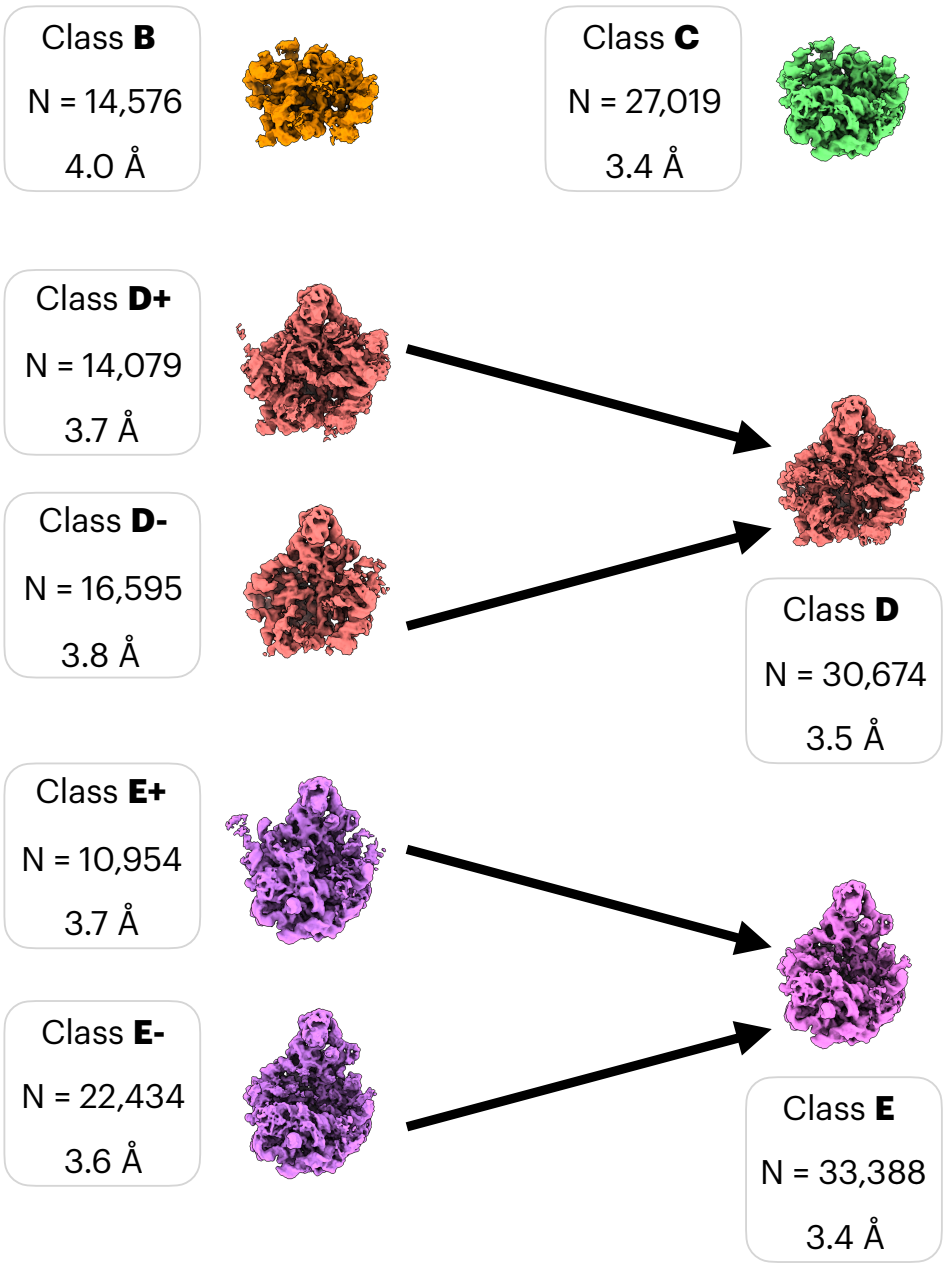


Figure 11: The major classes obtained through the reconstruction ensemble analysis of the 50S dataset. Class E features both the base (bottom) and the central protuberance (CP, top), Class D features only the CP, class C features only the base, and class B features none.

The EMPIAR-10076 entry is a single-particle cryo-EM dataset, consisting of the assembly intermediates of bL17-depleted *E. coli* large ribosomal subunit (LSU) (Davis et al., 2016). Specifically, this dataset consists of a picked particle stack ($N = 132$k, $D = 320$px, $1.31\mathring{A}$/pix), including 4 major classes of LSU assembly intermediates (B, C, D, E), 2 rare classes (the small 30S subunit F, and the complete 70S ribosome A), and 1 class of junk particles. This dataset has been commonly used as a benchmark dataset for heterogeneous reconstruction and analysis methods.

The canonical setup of this benchmark dataset follows the cryoDRGN (Zhong et al., 2021a) filtering to reduce the stack to $N = 97$k by eliminating the junk class prior to the heterogeneity analysis. The merit of the benchmarked method, then, is determined by its success in isolating the 4 major classes.

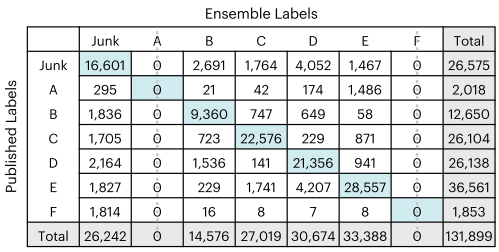


Figure 12: Confusion matrix between the published major class annotations and the predicted ensemble labels.

In order to provide stronger evidence of practical applicability of our method, we expand on this setup by processing the whole unfiltered stack ($N = 132$k), directly downloadable from EMPIAR. The major state classification is done *ab initio*, i.e. without prior pose information, in our setup. This is a more challenging setup than the known-pose/consensus volume setup used for models such as Zhong et al. (2021a) and Chen & Ludtke (2021).

In our major state classification (Figures 5, 11, and 12), we reach 75% accuracy in matching the published major class annotations (Davis et al., 2016, Fig. 4), successfully isolating the dataset into 5 classes (4 major classes and the junk particle class).

At the clustering stage, we used 7 clusters with the hopes that the rare classes A and F could also be captured. Our method could not capture these rare states. However, the clusters found included a subclassification of the classes D and E, which are named with $+$ and $-$ (Figure 11). The $+$ variants feature density in the L1 stalk (left in the provided projections) and shoulder (right in the provided projections) domains, while the $-$ variants do not feature L1 stalk. This finding was consistent with the minor classifications reported by Davis et al. (2016). These variants were merged to produce the 5-class classification, including the junk class (not pictured). The half-map FSC resolution and other map quality metric curves of each obtained class are reported through Figures 13-14.

In obtaining the result, our ensemble includes the following jobs, which are also listed on Table 5:

- 3 random replicates each of, 2-level hierarchies (root and children jobs, i.e. $K + 1$ jobs total in a tree), of $K = \{4, 8, 16\}$.
- 3 random replicates each of, 3-level hierarchy (root, children, and grandchildren jobs, i.e. $K^2 + K + 1$ jobs total in a tree), of $K = 6$.

Table 5: Job ensemble specification for the 50S analysis. The total number of jobs was 222, total GPU-hours used was 248.8, and the longest critical path was 4.3 hours.

# Replicates	K	Levels	Total Job Count	Total GPU-hours	Critical path (wall-hours)
3	4	2	15	8.0	1.1
3	6	3	129	103.2	2.4
3	8	2	27	28.8	2.1
3	16	2	51	108.8	4.3
			222	248.8	4.3

Assuming scalable compute access (e.g., cloud instances), the wall time could be as convenient as 4.3 hours. In our cryoSPARC setup, we had access to ≈ 50 parallel GPU job submission lanes, therefore our experiments were done overnight. Finally, the aggregation hyperparameters used were:

$$(\sigma, \beta, \gamma, p^{\min}, n_{\text{nbr}}) = (0.45, 5, 10, 1/450, 400) \tag{10}$$

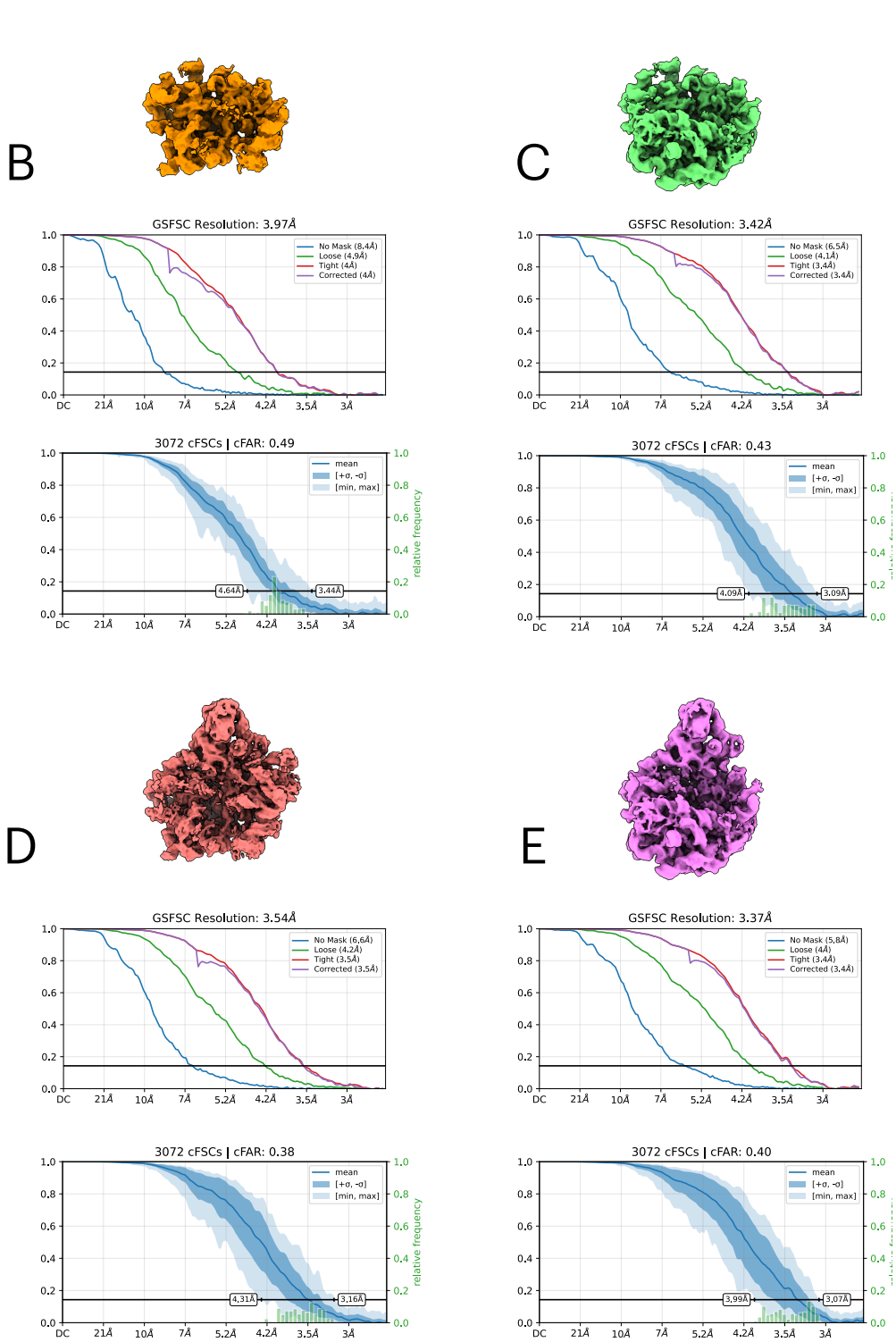


Figure 13: Fourier Shell Correlation (FSC) and conical FSC (cFSC) curves reported by cryoSPARC for classes B, C, D, and E.

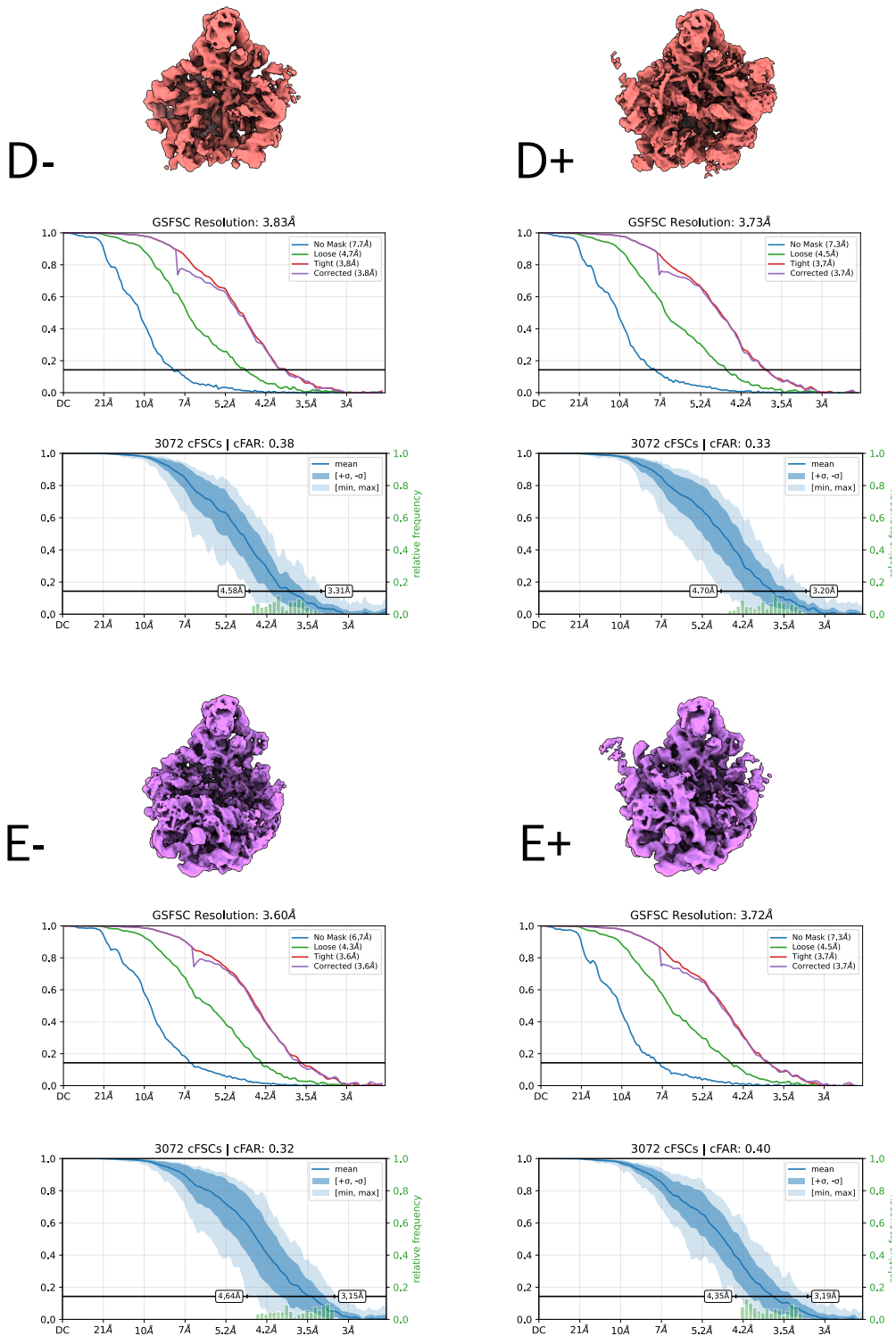


Figure 14: Fourier Shell Correlation (FSC) and conical FSC (cFSC) curves reported by cryoSPARC for classes D-, D+, E-, and E+.

F HYPERPARAMETER TUNING STUDY

We pursue a two-stage approach in our 5-dimensional hyperparameter grid sweep for alleviating the curse of dimensionality. We split the hyperparameters into two sets: the exponential factors (σ, β, γ) in the weight heuristics, and the other two $(p^{\min}, n_{\text{nbr}})$, namely the tiny class fraction threshold and the number of neighbors to keep in the sparsification process. In the following discussion, whenever $\%$ is used with respect to accuracy variation, it represents absolute percentage points and not relative variations.

We start with $(\sigma, \beta, \gamma) = (0.45, 5, 10)$ and conduct a grid sweep on $(p^{\min}, n_{\text{nbr}})$, reported in Figure 15. The results show negligible dependency on the tiny class proportion (at most 4.4 percentage points at a fixed n_{nbr}), but a crucial dependency on the sparsification threshold (up to 43.4 percentage points). Specifically, we observe that sparsifying the dense matrix up to only a few hundreds of neighbors remaining is important. Note that this is encouraging from a computational complexity perspective: the more the similarity matrix is sparsified, the less time and memory complexity the clustering methods would require (reduction of complexity from quadratic to linear in terms of the number of particles).

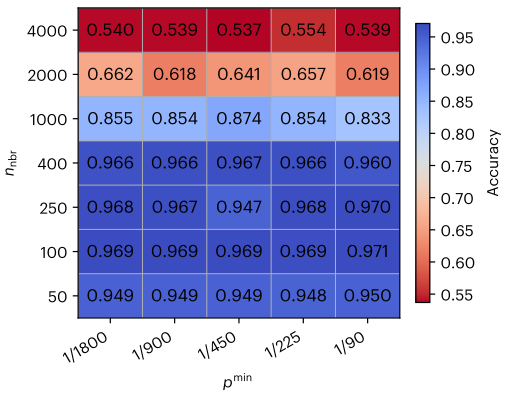


Figure 15: Accuracies obtained in the hyperparameter grid sweep for the tiny class proportion threshold $p^{\min}$ and the number of nearest neighbors kept n_{nbr} in the sparsification.

In the second phase of the hyperparameter search, we start with the result of the first sweep and explore (σ, β, γ). Specifically, we pick the best cell from above, i.e. $(p^{\min}, n_{\text{nbr}}) = (1/90, 100)$. The results are demonstrated in Figure 16. We observe that the fluctuations are limited to a band of 10%, highlighting the robustness of the weight heuristics under no tuning, while showing upside potential in tuning for minor performance gains in more detailed analyses.

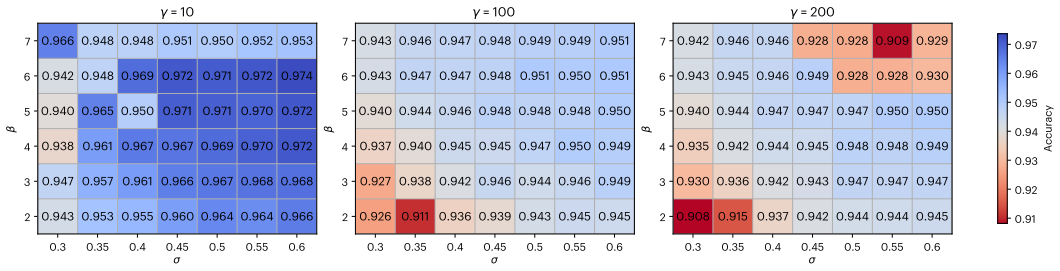


Figure 16: Accuracies obtained in the hyperparameter grid sweep for the exponential rate factors (σ, β, γ) in the weight heuristics.

For the results reported in Table 1, for both Tomotwin-45-asym and Tomotwin-100, the set of best performing hyperparameters above has been used:

$$(\sigma, \beta, \gamma, p^{\min}, n_{\text{nbr}}) = (0.6, 6, 10, 1/90, 100) \tag{11}$$

The job ensembles used in this study are the same as in the main results, therefore requiring no further GPU-hours. The primary driver of the computational complexity in this hyperparameter sweep is the clustering step. Each clustering assignment takes a few minutes on a single CPU core, hence using parallel CPUs, the ablation study is done in an hour.