Article

Bayesian Inference of Initial Models in Cryo-Electron Microscopy Using Pseudo-atoms

Paul Joubert[1,*] and Michael Habeck[1,2,*]

[1]Felix-Bernstein Institute for Mathematical Statistics, Georg-August-Universität Göttingen, Göttingen, Germany; and [2]Max Planck Institute for Biophysical Chemistry, Göttingen, Germany

ABSTRACT Single-particle cryo-electron microscopy is widely used to study the structure of macromolecular assemblies. Tens of thousands of noisy two-dimensional images of the macromolecular assembly viewed from different directions are used to infer its three-dimensional structure. The first step is to estimate a low-resolution initial model and initial image orientations. This is a challenging global optimization problem with many unknowns, including an unknown orientation for each two-dimensional image. Obtaining a good initial model is crucial for the success of the subsequent refinement step. We introduce a probabilistic algorithm for estimating an initial model. The algorithm is fast, has very few algorithmic parameters, and yields information about the precision of estimated model parameters in addition to the parameters themselves. Our algorithm uses a pseudo-atomic model to represent the low-resolution three-dimensional structure, with isotropic Gaussian components as moveable pseudo-atoms. This leads to a significant reduction in the number of parameters needed to represent the three-dimensional structure, and a simplified way of computing two-dimensional projections. It also contributes to the speed of the algorithm. We combine the estimation of the unknown three-dimensional structure and image orientations in a Bayesian framework. This ensures that there are very few parameters to set, and specifies how to combine different types of prior information about the structure with the given data in a systematic way. To estimate the model parameters we use Markov chain Monte Carlo sampling. The advantage is that instead of just obtaining point estimates of model parameters, we obtain an ensemble of models revealing the precision of the estimated parameters. We demonstrate the algorithm on both simulated and real data.

INTRODUCTION

Single-particle cryo-electron microscopy (cryo-EM) is a method used to determine the three-dimensional structure of macromolecular assemblies (1). Many copies of the assembly of interest are prepared in a thin ice layer, and imaged using an electron microscope. Each image, called a micrograph, contains non-overlapping two-dimensional images of hundreds of particles, all assumed to have approximately the same three-dimensional structure, but oriented differently. Tens of thousands of these particle images are extracted from a collection of micrographs. Such large numbers are required due to the extremely low signal-to-noise ratio (SNR) of the images.

The standard image formation model for this setting is to model each image as the linear projection of the unknown structure along an unknown direction, convolved with a known point-spread function (due to the electron microscope), and corrupted by noise (1). The reconstruction problem is to infer the three-dimensional structure from the two-dimensional images.

The workflow for solving the reconstruction problem can be divided into two parts: obtaining a low-resolution initial model, followed by a refinement of this model. A common refinement algorithm is three-dimensional projection matching (2). It alternates between updating the three-dimensional model and the image orientations. Given the current three-dimensional model, its projections are calculated along a discrete grid of directions. Each image is aligned to the best matching projection. Having updated all the image orientations, a new three-dimensional model is reconstructed using direct Fourier inversion for example, and the steps are repeated until convergence.

A low-resolution initial model could be a previous reconstruction of the same structure, or a model of a similar structure. However, in cases where a suitable initial model is not available, it has to be reconstructed from the data using an ab initio reconstruction algorithm. This is an important step: if the initial model does not represent the structure accurately enough, it may lead the refinement algorithm to converge to an incorrect model.

As input data, many ab initio algorithms do not use the individual particle images, but work with two-dimensional class averages instead. Class averages are obtained by clustering, aligning, and averaging the two-dimensional images to improve the SNR (3). This significantly reduces the number of unknown image orientations to be estimated by the algorithm.

Several ab initio algorithms exist. They include common-lines-based algorithms (4–9), random-model methods (10,11), methods using stochastic hill climbing (12) or

Submitted August 14, 2014, and accepted for publication December 23, 2014.

*Correspondence: pjouber@gwdg.de or mhabeck@gwdg.de

Editor: Dorit Hanein.

0006-3495/15/03/1165/11 $2.00

http://dx.doi.org/10.1016/j.bpj.2014.12.054

nonlinear dimensionality reduction (13), and a Bayesian approach (14).

A drawback of most of these ab initio algorithms is that they have many ad hoc parameters whose effect on the results is difficult to interpret, and which are a potential source of bias by the user.

This has motivated the use of statistical modeling in cryo-EM reconstruction, first in the form of maximum-likelihood methods (15,16), and more recently as maximum a posteriori (MAP) estimates in a Bayesian framework (14,17,18). Statistical modeling requires a complete description of how the data (i.e., the two-dimensional images) are generated from the model (i.e., the three-dimensional structure and image orientations). It distinguishes between parameters used to describe the statistical model and algorithmic parameters influencing, for example, how fast the algorithm runs, but which cannot bias the results. Such an approach therefore has parameters that are easier to interpret, and a higher degree of objectivity.

The cryo-EM reconstruction problem is highly ill-posed: different models can give rise to very similar data. The standard way of dealing with this is to regularize, for instance by penalizing three-dimensional structures with too much high-frequency content. From the Bayesian perspective, this is equivalent to introducing prior assumptions about the model (in this case that the three-dimensional structure should have mostly low-frequency content). The Bayesian approach provides a systematic and theoretically well-grounded way to combine such explicit prior knowledge about the model with the data to find models (i.e., three-dimensional structures and image orientations) that are consistent with both the prior knowledge and the data.

Bayesian approaches to reconstruction algorithms tend to be very computationally intensive. Typical computation times in CPU time range from days (14) to several months (15).

We introduce a probabilistic ab initio algorithm that addresses the above-mentioned challenges. It uses a pseudo-atomic model with several hundred pseudo-atoms that can move around and change their size. As we will show later, this significantly reduces the number of parameters needed to describe the three-dimensional structure. Computing two-dimensional projections of the three-dimensional structure also becomes much simpler and faster.

Our reconstruction algorithm uses a Bayesian approach. The data-generation process is simple and intuitive, with only a small number of adjustable parameters such as the number of pseudo-atoms. Expressing prior knowledge becomes straightforward.

Instead of just generating the single model most consistent with the data and prior knowledge (the MAP estimate), our algorithm generates multiple similar models that are all consistent with the data and prior knowledge. The ensemble of models can be analyzed to obtain information about the precision of the estimated three-dimensional structure and image orientations. This approach also allows us to integrate out the image orientations without the use of a discrete grid, which slows down other Bayesian approaches.

We demonstrate our algorithm using simulated and experimental data, and show that in all cases it can obtain suitable initial models in a relatively short time.

MATERIALS AND METHODS

Model

Pseudo-atomic model

We use a coarse-grained representation of the three-dimensional structure as a cloud of K pseudo-atoms. Each pseudo-atom is a spherical blob centered at position μ_k with unknown radius σ. All the pseudo-atoms have the same (adjustable) size, and their positions can vary continuously, i.e., they are not fixed to a regular grid. Each pseudo-atom has an unknown weight w_k. In analogy to high-resolution atomic structures, the μ_k vectors are the Cartesian coordinates of the kth pseudo-atom, and w_k and σ are its occupancy and temperature factor, respectively. In contrast to atomic models, pseudo-atoms are much larger than atoms, and far fewer of them are therefore required to represent a structure.

Pseudo-atomic models have been used to rigidly fit multiple subunits into a given low-resolution three-dimensional density map (19), and to identify possible conformational changes through a normal mode analysis (20,21). In all these applications the pseudo-atomic model is fit to a three-dimensional structure that has been reconstructed earlier using other algorithms. In contrast, we are exploiting the advantages of the pseudo-atomic representation for the reconstruction problem itself: the parameters of the pseudo-atoms will be estimated directly from the two-dimensional images, without any reference to three-dimensional volumes on regular grids.

If we choose our pseudo-atoms to have a Gaussian shape, then from a statistical perspective our representation is known as a Gaussian mixture model (GMM) (22). GMMs are widely used to estimate probability density functions from observed data points, and are a smooth and efficient alternative to the common histogram estimator. The advantage of casting cryo-EM reconstruction as a GMM fitting problem is that we can draw inspiration from well-established statistical methods for estimating the parameters of the pseudo-atoms.

Each pseudo-atom is represented by a Gaussian function $G_{3D}(x;\mu_k,\sigma)$ describing the density at the three-dimensional point x of a pseudo-atom centered at μ_k with radius σ (the parameters μ_k and σ are the mean and standard deviation of the Gaussian). The density map ρ representing the entire three-dimensional structure is a weighted sum of K such pseudo-atoms,

$$\begin{aligned}\rho(x) &= \sum_{k=1}^{K} w_k G_{3\mathrm{D}}(x;\mu_k,\sigma) \\ &= \sum_{k=1}^{K} \frac{w_k}{(2\pi)^{3/2}\sigma^3} \exp\left\{-\frac{\|x-\mu_k\|^2}{2\sigma^2}\right\},\end{aligned} \qquad (1)$$

where $||x - \mu_k||$ denotes the Euclidean distance between any three-dimensional point x and the position of the pseudo-atom μ_k. Equation 1 is used, for example, in many flexible fitting algorithms that fit atomic structures to experimental EM maps.

The pseudo-atomic model has many advantages compared to the standard three-dimensional grid based representation. The first advantage is that it requires far fewer parameters (see Fig. 1 and Movie S1 in the Supporting Material). Each pseudo-atom needs only four parameters to describe its position and weight, whereas a three-dimensional grid has one parameter for each three-dimensional voxel. The pseudo-atomic model can also be evaluated on an arbitrarily fine grid for visualization purposes.

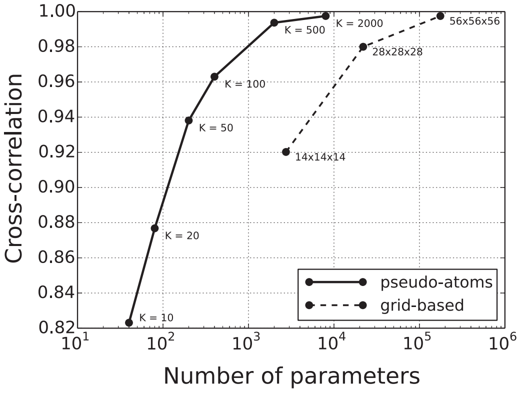

FIGURE 1 Comparison between the number of parameters required for the standard grid-based representation and the pseudo-atomic representation of RNA polymerase II. The normalized cross-correlation was computed with respect to the original reference structure on a grid with dimensions 112 × 112 × 112. The reference structure was downsampled by factors of 2, 4, and 8 to obtain the grid-based representations. The number of parameters is the number of voxels and four times the number of components (K) for the respective representations. The figure shows that, for any specified level of accuracy (quantified by the cross-correlation), the pseudo-atomic representation needs <10% of the number of parameters needed by the grid-based representation. See also Movie S1.

Another important advantage is that computing two-dimensional projections is simple and fast. A given image orientation is described by a three-dimensional rotation matrix R. The three-dimensional structure is projected along the corresponding direction by first rotating it by R, and then integrating along the z axis to obtain an image in the x,y plane. The in-plane translation is denoted by the vector t. In-plane rotations are already accounted for by the rotation matrix R.

Applying this procedure to our pseudo-atomic model is very simple: first we apply the rotation by transforming each pseudo-atom position μ_k to $R\mu_k$. Then we project to the x,y plane by just discarding the z coordinate. Formally, we project $R\mu_k$ to $PR\mu_k$, where

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Finally, we translate the projection by t. The resulting translated two-dimensional projection of all the pseudo-atoms is also a GMM, of the form

$$(PR\rho)(x) = \sum_{k=1}^{K} w_k G_{2\mathrm{D}}(x; PR\mu_k + t, \sigma), \tag{2}$$

where

$$G_{2\mathrm{D}}(x;\mu,\sigma) = \frac{1}{2\pi\sigma^2}\exp\left\{-\frac{\|x-\mu\|^2}{2\sigma^2}\right\}$$

is a two-dimensional Gaussian. The weights w_k and size σ are the same as for the three-dimensional model. The computation only requires a small number of elementary matrix operations. There is also no need for any interpolation.

The parameters of the pseudo-atomic model, together with a rotation R_i and translation t_i for each image, constitute our unknown model parameters,

$$\theta = \{\mu, \sigma, w, R, t\},$$

where $\mu = \{\mu_k\}$, $w = \{w_k\}$, $R = \{R_i\}$, and $t = \{t_i\}$.

We do not have to specify the size σ of the pseudo-atoms because it is estimated by the algorithm. Instead, we have to specify the number of pseudo-atoms, K, which implicitly determines the optimal size σ. We choose K such that the estimated size σ is approximately the same as the pixel size of the two-dimensional projection images. A similar rule-of-thumb has been shown to work when fitting atomic models to three-dimensional volumes (20), by choosing the pseudo-atom size to be roughly the same as the voxel size.

Using the Bayesian approach, we have to encode our prior assumptions by defining a probability distribution $p(\theta)$ (called the "prior") over all possible models describing how plausible each model is before including any data. For instance, we use a three-dimensional Gaussian distribution as the prior for each position μ_k to encode our assumption that the pseudo-atoms should be spread across a region roughly the size of the unknown three-dimensional structure. For each rotation R_i we use a uniform prior to model the assumption that each image orientation is equally likely.

Our distribution of the prior has only four additional parameters (hyperparameters) that determine the shape of the prior, and are fixed during the reconstruction: one for the expected size of the structure (which can be estimated from the size of the images), one determining how much the individual weights w_k are allowed to deviate from the average weight, and two specifying the range of plausible sizes σ for the pseudo-atoms. The hyperparameters have only a minor influence on the final model, and the default values will work for a large number of reconstructions. See the Supporting Material for an analysis of the effect of the hyperparameters on the final model.

Data-generation model

One way to create two-dimensional projection images is to project the three-dimensional mixture model to a two-dimensional mixture model as described above (Eq. 2), and then to evaluate this two-dimensional mixture model on a two-dimensional grid. This approach will be used below to generate simulated data.

Viewing our pseudo-atomic model as a GMM, we would like to make use of the powerful statistical algorithms that exist for fitting GMMs to three-dimensional point clouds. Examples of such algorithms include expectation maximization (23) and Gibbs sampling (24). Two complications prevent us from directly applying one of these algorithms: we have two-dimensional intensities instead of individual two-dimensional points, and a dimension is missing (we have two-dimensional data instead of three-dimensional points). To address these complications and connect with existing methods for estimating GMM parameters, we adopt an alternative view of the data-generation process (see the Supporting Material for a formal description).

Starting with a pseudo-atomic structure, we assume that the first step in generating the ith image is to generate a three-dimensional point cloud with C points covering the same region as the pseudo-atomic model. Each point in the point cloud is created by first randomly selecting a pseudo-atom according to its weight w_k, and then randomly placing the point near the center of the pseudo-atom. This is exactly how algorithms such as expectation maximization and Gibbs sampling assume the three-dimensional point cloud to have been generated in the standard application of fitting a GMM to a three-dimensional point cloud.

The three-dimensional point cloud is then rotated and translated by R_i and t_i, and projected to a two-dimensional point cloud by discarding the z coordinate. Finally, a two-dimensional histogram is formed by using the two-dimensional pixels as bins. The two-dimensional histogram is viewed as a quantized image, with the number of points in each bin being the image intensity at that pixel. The input data $\mathcal{D}$ to the algorithm consists of all the quantized images together.

The randomness in generating the three-dimensional point cloud translates into randomness in the generated data $\mathcal{D}$. Given fixed parameters for the pseudo-atoms and the rotations and translations, the probability distribution over possible datasets that can be generated is denoted by $p(\mathcal{D} \mid \theta)$. In statistical parlance, $p(\mathcal{D} \mid \theta)$ is called the "data likelihood" and defines a random forward model of how the observed images could have been generated. Estimation of the model parameters θ is achieved by inverting the data generation process with the help of Bayes' theorem.

To use our ab initio algorithm we first have to convert the raw particle images to quantized class averages. The preprocessing steps needed to obtain nonnegative (real-valued) class averages are described below. The nonnegative images are converted to quantized images by first scaling them by α and then rounding to the nearest integer. The scaling factor α is chosen such that the total number of points C in the image equals a predetermined constant.

The idea of the ab initio algorithm to be described below is to reverse the above data-generation process: starting with two-dimensional points from the quantized image, we first back-project them to three-dimensional points by estimating their missing z coordinates. Then we assign each three-dimensional point to a pseudo-atom that was likely to have generated it. And finally we move the pseudo-atom to align it to its assigned three-dimensional points. The last part of this strategy is the same as used in the standard application of Gibbs sampling to three-dimensional point clouds, and very similar to expectation maximization.

Data preprocessing

Similarly to many other ab initio algorithms, we use class averages instead of raw particle images as input to our algorithm. This yields a computational advantage by significantly reducing the number of unknown rotations, in addition to an increase in the SNR. It comes at the cost of corrupting the high-frequency information in the images, but this is not a drawback for ab initio methods, where we are only interested in low-resolution reconstructions.

Our ab initio algorithm requires the class averages to be nonnegative. This is a sensible assumption, given that in the standard model of cryo-EM image formation, the images are taken to be nonnegative before the contrast transfer function (CTF) is applied. If we apply commonly used CTF-correction algorithms such as Wiener filtering or phase-flipping, the resulting images typically still have negative values.

Here we describe an extra deconvolution step to correct for the CTF that can be appended to the class-averaging algorithm to ensure that the resulting images are nonnegative (see Fig. 2). The deconvolution algorithm can be applied either to individual raw images that have been clustered and aligned, or to class averages as produced by any existing class-averaging algorithm.

Let $\{z_i\}$ be the raw images from a single class that have been two-dimensionally aligned relative to each other. We model each image $z_i = f_i * y + \epsilon_i$ as the convolution of a nonnegative image y with a point-spread function f_i, with added independent and identically distributed Gaussian noise ϵ_i. Each point-spread function is the inverse Fourier transform of the corresponding CTF for that image, which is assumed to be known. The unknown image y is the projection of the unknown density map along an unknown direction.

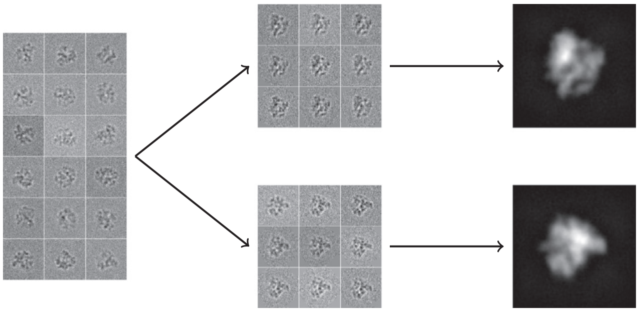

FIGURE 2 Preprocessing pipeline to prepare the data for the ab initio algorithm. The raw images on the left are clustered and aligned using any of the standard class-averaging algorithms. The deconvolution algorithm in the text is then applied to every cluster or class average to obtain a deconvolved image (on the *right*).

A MAP estimate for y is found by minimizing the convex loss function,

$$L(y) = \frac{1}{2}\sum_i \|z_i - f_i * y\|^2 + \frac{1}{2}\alpha\|\nabla y\|^2, \quad (3)$$

subject to the constraint that y be nonnegative. Here ∇ is the gradient operator, and α is a hyperparameter controlling the smoothness of y. The regularization parameter α can be chosen manually, and was fixed to a value of 10 for all our experiments. We use the L-BFGS-B algorithm (25) to optimize Eq. 3.

Algorithm

In the previous section we defined the model parameters θ, the data $\mathcal{D}$, the prior distribution on the parameters $p(\theta)$, and the data-generation model $p(\mathcal{D} \mid \theta)$. Bayes' theorem dictates how to compute the posterior distribution $p(\theta \mid \mathcal{D})$:

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})}. \quad (4)$$

The posterior is a probability distribution over all possible models, quantifying how well each model explains the data without violating the prior assumptions.

Gibbs sampling is a widely-used Markov chain Monte Carlo algorithm for sampling from the posterior distribution, in other words for generating model realizations that follow the posterior distribution and are therefore consistent with both the data and the prior information (24).

The first step is to generate a random initial model by sampling the model parameters from the prior distribution. The parameters are then updated in turn: first, the assignments of points to pseudo-atoms and missing z coordinates (back-projection); then, the pseudo-atom parameters (positions, weights, and size); and finally, the rotations and translations. Each parameter update depends on the current values of the other parameters. This single Gibbs sampling step is iterated several times until the parameters converge to a stable region of parameter space that should be independent of the initial random model.

Each group of parameters is updated according to their corresponding conditional distribution. The conditional distribution quantifies the likelihood of each possible value of a given parameter, assuming that all other model parameters are known and fixed.

Importantly, in the Bayesian framework all the conditional distributions are completely determined by just the prior distribution $p(\theta)$ and the data-generation model $p(\mathcal{D} \mid \theta)$. The only way to modify these distributions is by making different prior assumptions or by using different data. Furthermore, each conditional distribution is a well-known distribution for which is it straightforward to generate parameters. For instance, the conditional distribution for each pseudo-atom position is a Gaussian distribution. Less well known is the conditional distribution for each rotation R_i, which is of the form $\exp[\mathrm{tr}(A^T R_i)]$ for some matrix A. This is a unimodal distribution, which can be seen as the analog for three-dimensional rotations of the well-known von Mises distribution for two-dimensional rotations. We use the algorithm introduced by Habeck (26) to generate rotations from this distribution.

We will first give an overview of the entire algorithm, which consists of several Gibbs sampling steps, and then describe a single Gibbs sampling step in more detail.

In the flowchart in Fig. 3, the algorithm is divided into two parts: an initial stage and a refinement stage. A very low resolution structure with only 100 or 200 pseudo-atoms is constructed during the initial stage, and then refined with 500 or 2000 pseudo-atoms during the refinement stage. See Movie S2 for a visualization of the algorithm.

The initial stage is reminiscent of the projection-matching algorithm described in the Introduction. We alternate between multiple Gibbs sampling steps to update just the pseudo-atom parameters, and multiple Gibbs

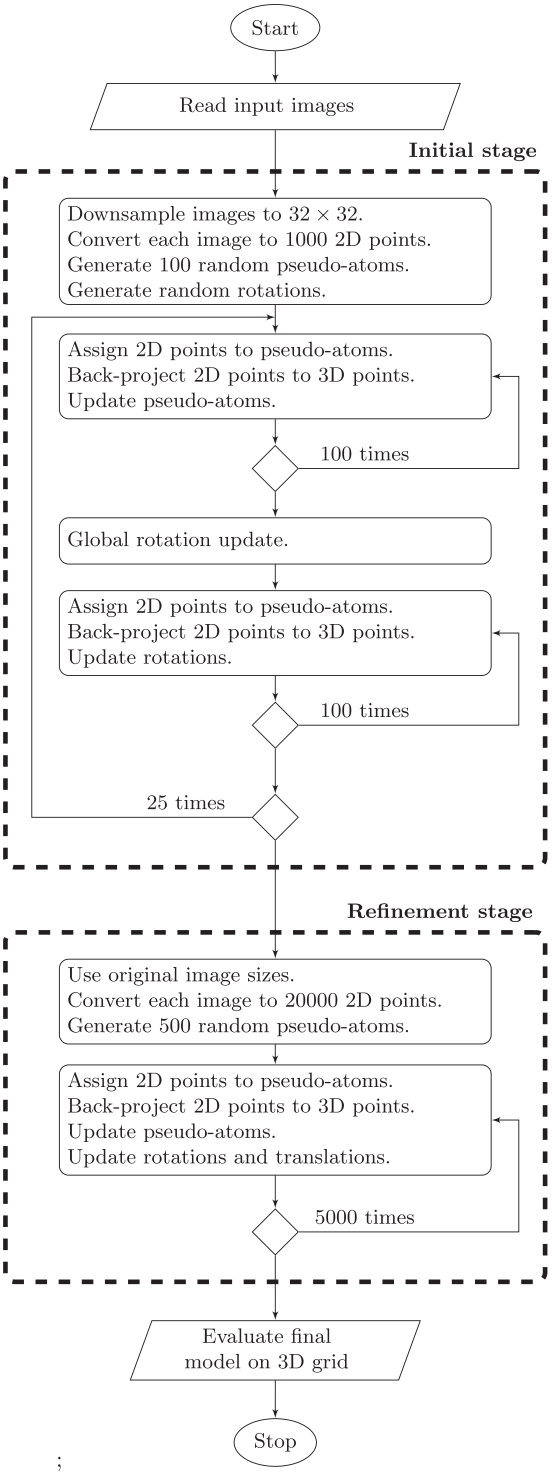
Start
Read input images
Initial stage
Downsample images to 32 × 32.
Convert each image to 1000 2D points.
Generate 100 random pseudo-atoms.
Generate random rotations.
Assign 2D points to pseudo-atoms.
Back-project 2D points to 3D points.
Update pseudo-atoms.
100 times
Global rotation update.
Assign 2D points to pseudo-atoms.
Back-project 2D points to 3D points.
Update rotations.
100 times
25 times
Refinement stage
Use original image sizes.
Convert each image to 20000 2D points.
Generate 500 random pseudo-atoms.
Assign 2D points to pseudo-atoms.
Back-project 2D points to 3D points.
Update pseudo-atoms.
Update rotations and translations.
5000 times
Evaluate final
model on 3D grid
Stop
;

FIGURE 3 Algorithm flowchart showing initial and refinement stages. The initial stage consists of 25 steps of updating first the pseudo-atoms, then the rotations, each using 100 Gibbs sampling steps. The refinement stage consists of 5000 Gibbs sampling steps, of which the first 2500 are discarded as the burn-in phase. The number of steps is conservatively chosen to be far more than is needed for the algorithm to converge in all tested cases.

sampling steps to update just the rotations. The rotation updates using Gibbs sampling tend to make only small adjustments, and can sometimes get stuck in local optima. Therefore we add a global rotation update during every outer loop. During this global rotation update, each image is compared to 10,000 two-dimensional projections of the pseudo-atomic model in random orientations. For each orientation, we compute the likelihood of the image given the two-dimensional projection corresponding to the orientation. These likelihoods form the coefficients of a discrete approximation to the conditional posterior distribution over the rotation. We then sample an orientation from this discrete distribution, and use it to update the rotation for the image. In this way rotations can escape local optima.

For the refinement stage, we increase the number of pseudo-atoms to 500 or 2000 and then sample all parameters using Gibbs sampling. During this stage, only minor adjustments are made to the rotations.

During both stages we can monitor the progress and convergence of the Gibbs sampler via the log-posterior, defined as log $p(\theta \mid \mathcal{D})$, where θ is the model being used (see Fig. 5 *B*).

The output of the algorithm is an ensemble of pseudo-atomic models from the posterior distribution. This ensemble consists of every 50th model generated during the refinement stage after discarding the first 2500 models to exclude the burn-in period. To represent the result as a single volume we evaluate each model in the posterior family on a three-dimensional grid, and report either the mean of these volumes, or any one of the volumes (they are all very similar).

We explain a single Gibbs sampling step in Fig. 4 using a toy two-dimensional reconstruction example with only two pseudo-atoms and two images. Each one-dimensional image is shown as a bar chart with the height of each bar indicating the number of points at the corresponding one-dimensional pixel. For clarity, most of the steps are shown only for the data lying below the pseudo-atoms.

The first step (panels *2* and *3*) is to evaluate the one-dimensional projection of each pseudo-atom at all the one-dimensional pixels, and assign points to pseudo-atoms. At each pixel, the relative value of the two pseudo-atoms determines the proportion of points to assign to each. For instance, for pixels on the left, all points are assigned to the bottom-left pseudo-atom, while for pixels in the middle, the points are distributed equally among the two pseudo-atoms (only half the bar is shaded). The second step (panel *4*) is to estimate the missing y coordinates (missing z coordinates in the three-dimensional case). For each one-dimensional point, its y coordinate is chosen randomly near the y coordinate of the pseudo-atom to which it was assigned. The next step (panel *5*) is to update the pseudo-atoms, i.e., their weights, positions, and size.

In this example, we update only their positions. The position of each pseudo-atom is chosen randomly near the mean of the two-dimensional points assigned to that pseudo-atom. After this update, the one-dimensional projections of the new pseudo-atomic model match the input data more closely. In the final step (panel *6*), we update the rotations. The pseudo-atoms remain fixed, and the two-dimensional point cloud is rotated about the origin to better fit the pseudo-atoms. As a result of the rotation, the one-dimensional projections match the input data very well.

RESULTS

We used five different datasets to test our algorithm: one consisting of simulated class averages, one with realistically simulated raw particles, and three with real data.

For the first dataset, we converted an atomic model of the ribosome 50S subunit (Protein Data Bank (PDB): 1VOR) to a three-dimensional volume at 15 Å using the software CHIMERA (University of California at San Francisco, San Francisco, CA) and projected it using random orientations to create 25 class averages. The size of the images is 50 × 50 pixels and the sampling rate is 6 Å/pixel.

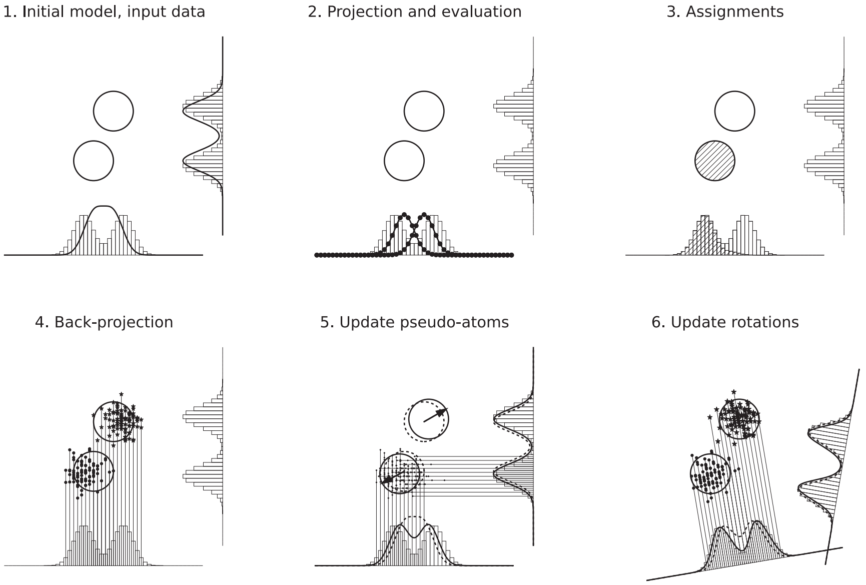

FIGURE 4 Simple two-dimensional reconstruction example to explain a single Gibbs sampling iteration. (*Solid lines*) One-dimensional projections of this model. (*Dashed lines*) One-dimensional projections of the previous model. Initially (*1*), the one-dimensional projections differ significantly from the one-dimensional data. They improve after moving the pseudo-atoms (*5*), and once again after updating the image orientations/rotations (*6*). The final projections approximate the data quite well.

Fig. 5 *A* shows the progression of the reconstruction, including the positions of the pseudo-atoms. Also see Movie S2. The computation took 22 min on a single core, with 7 and 15 min for the initial and refinement stages, respectively. We used 100 pseudo-atoms for the initial stage and 2000 for the refinement stage. The models produced by the refinement stage have a pseudo-atom size of ~5.0 Å, as can be seen in Fig. 5, *B* and *C*. To evaluate the reconstruction, we created a reference volume at $res = 25$ Å. The reference volume was created from the atomic model with CHIMERA's MOLMAP command, which describes each atom using a three-dimensional Gaussian with size $0.225 \times res \approx 5.6$ Å, i.e., almost the same as our final pseudo-atom size. Our final reconstruction agrees very well with the reference: the normalized cross-correlation between the two structures is 0.990, and they agree to a resolution of 15.9 Å as measured by the $FSC = 0.5$ criterion (27). The FSC curves are shown in Fig. S1 in the Supporting Material. We also compared our final estimated rotations with the true rotations, and found that most of them agree to within 0.5° with a maximum error of $<1.5°$ (see Fig. 5 *D*).

The second dataset consists of 5000 realistically simulated RNA polymerase II (PDB: 1I3Q) particles with size 100 × 100 pixels at a sampling rate of 2.5 Å/pixel. The reference volume was projected along random orientations, and random translations were applied to the images. The CTF was applied with a random defocus value for each image, followed by Gaussian noise with $SNR = 0.2$ (Fig. 6). We used EMAN2 (28) to compute 41 class averages in 98 min, followed by deconvolution, which took 73 min. Applying our ab initio algorithm to the deconvolved images took another 38 min, a total of 209 min. The final reconstruction has a cross-correlation of 0.966 compared to the reference model at 20 Å, and they agree to a resolution of 14.5 Å at $FSC = 0.5$.

For the third dataset, we used publically available experimental 70S ribosome data from the EMDB test image data (15,29). The dataset consists of 5000 images with size 130 × 130 at a sampling rate of 2.82 Å/pixel. We used the software toolbox ASPIRE (30) to compute 50 class averages, followed by deconvolution. The algorithm was initialized with GroEL, an unrelated structure. Fig. 7 shows how the algorithm eliminates the bias caused by the incorrect initial model and quickly converges to the correct 70S structure. To provide further evidence of the robustness of the algorithm, we successfully repeated the reconstruction with a random initial model. We compared the final reconstruction to the result obtained using the PRIME algorithm (12). As shown in Fig. 7, the two structures are visually very similar, certainly enough for each to be used as initial model for a refinement. The normalized cross-correlation between the two structures is 0.900, and they agree to a resolution of 31.1 Å at $FSC = 0.5$. The computation of the first reconstruction (starting with GroEL) took 102 min in total (50 min for forming class averages, 24 min for deconvolution, and 28 min for the initial and refinement stages of our algorithm). The class-averaging step used eight cores on a desktop computer, while the other steps used a single core on a laptop, a total of <8 CPU h. In contrast, the PRIME reconstruction took ~10 h on a cluster with 40 cores. In general, PRIME takes ~500–1000 CPU hours to compute an initial model. This example shows that our algorithm produces comparable results in a fraction of the time required by PRIME.

For the fourth experiment, we used a publically available experimental GroEL dataset (31) consisting of ~5000 images with size 128 × 128 at a sampling rate of 2.12 Å/pixel. EMAN2 was used to obtain 13 class averages in 19 min, followed by deconvolution, which took 2 min. Applying the

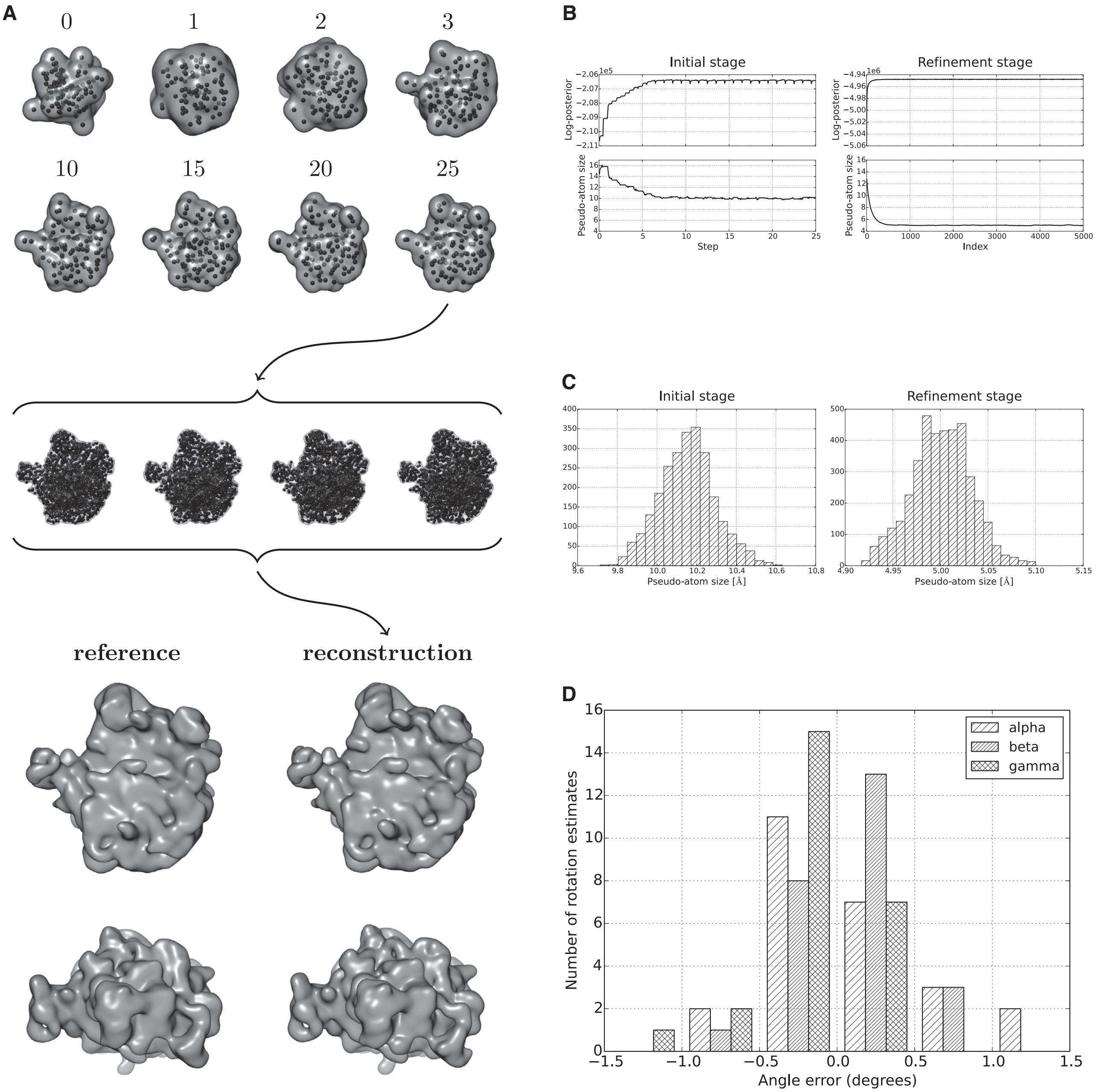

FIGURE 5 Results for the 50S ribosome. (*A*) Starting from a random initial model, the initial stage converges within <10 steps. The number of pseudo-atoms (shown as *small solid circles*) is then increased from 100 to 2000, and multiple models from the posterior distribution are shown. These are averaged to obtain the final reconstruction. The cross-correlation with the reference model at 25 Å is 0.990. (*B*) The log-posterior measures how well the estimated model matches the data and the prior. It shows that both the initial and refinement stages converge rapidly. The estimation of the optimal pseudo-atom size converges fast as well. The figure shows that increasing the number of pseudo-atoms leads to a decrease in their size from ~10 to ~5 Å. (*C*) Instead of a single value for the pseudo-atom size, the algorithm gives us a distribution of plausible sizes. Comparing the distributions shows that the size is more precisely determined for the refinement stage. (*D*) For each of the 25 images, we compare the Euler angles of the original rotation and the final rotation. All rotation estimates are very accurate, with most of the angular errors <1°.

initial and refinement stages of our algorithm took another 13 min, for a total of 34 min. We also took into account the known D7 symmetry of the structure, which shows that our framework is flexible enough to include symmetry constraints as prior information. Our final result (Fig. 8) has a cross-correlation of 0.927 with the reference model (PDB:1OEL) at 20 Å, and they agree to a resolution of 17.5 Å at $FSC = 0.5$.

For the fifth and final experiment, we tested the algorithm using experimental data from the human Anaphase Promoting Complex (APC/C) (32). Approximately 10,000 particles of size 80 × 80 pixels at a sampling rate of 4.9 Å/pixel were

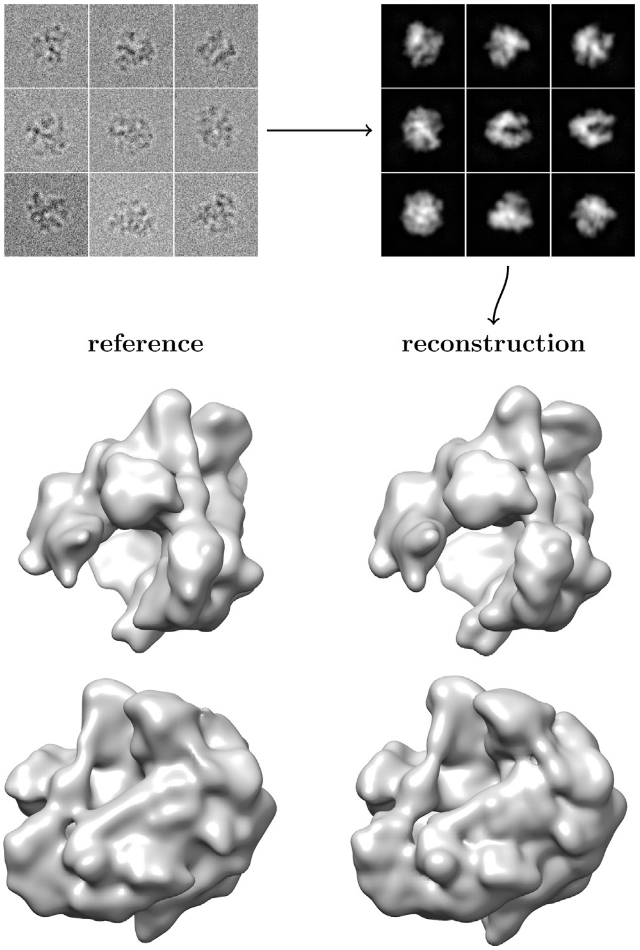

FIGURE 6 Results for realistically simulated RNA polymerase II data. At the top left are nine of the 5000 raw particles that were used to compute 41 deconvolved class averages, of which nine are shown (*top right*). The final reconstruction agrees well with the reference at 20 Å, as shown by the cross-correlation value of 0.966.

processed using reference-free alignment to produce 61 class averages. As required for a realistic test case, no knowledge of previous structures was used in computing the class averages. As before, the class averages were deconvolved to obtain nonnegative images, which were then used as input to our ab initio algorithm. Our reconstruction (Fig. 9) was compared to the reconstruction (EMD-2354) published earlier using data from the same source (32). The structures have a cross-correlation of 0.902 and agree to a resolution of 24.8 Å at $FSC = 0.5$.

DISCUSSION

Our algorithm differs significantly from other cryo-EM reconstruction algorithms in the way in which three-dimensional structures are represented. Typically one uses a cubic three-dimensional grid comprising a large number of voxels. An alternative approach is to use rotationally symmetric blobs (33), each roughly the size of a voxel, positioned on a regular three-dimensional grid. The blobs are fixed in shape and size, the only free parameters being their weights. Thus the voxel and blob representations require a similar number of parameters, but projecting from three dimensions to two dimensions is faster and more accurate when using blobs instead of voxels. Blobs are used in the software package XMIPP (34), and were reported to produce superior quality reconstructions at lower computational cost (33).

Our approach can be seen as an extension of the blob approach, where we use pseudo-atoms instead of blobs, and allow their positions to vary smoothly instead of fixing them to a regular grid. This allows for a more parsimonious representation (Fig. 1), as pseudo-atoms can be moved to regions where they are more needed. Furthermore, the size of voxels or blobs needs to be fixed before reconstruction. In our case, the pseudo-atoms still all have the same size, but the appropriate size is estimated during reconstruction (Fig. 5). Instead of specifying their size, we have to choose the number of pseudo-atoms. There is a strong inverse relation between the number of pseudo-atoms and their size: as the number increases, the size must decrease to fill the same volume. Therefore, choosing the number is equivalent to implicitly choosing the size.

As mentioned before, our rule-of-thumb is to choose the number of pseudo-atoms such that the resulting pseudo-atom size is similar to the pixel size. For example, for our ribosome reconstruction, during the initial stage the pixel size is 9.4 Å and the final pseudo-atom size is ~9.9 Å, indicating that 100 pseudo-atoms was an appropriate choice. For the refinement stage, the corresponding values are 6.0 Å and 5.0 Å. Guided by this strategy, we used either 100 or 200 pseudo-atoms for the initial stage, and either 500 or 2000 pseudo-atoms for the refinement stage, for all our experiments.

The significant reduction in the number of parameters needed to describe a structure (Fig. 1) has two advantages. The first is that the algorithm is very fast. Starting from the class averages, the algorithm took <40 min for each of the five structures. All experiments with the exception of class-averaging with ASPIRE were done on a standard laptop (Dell, Round Rock, TX) with a 2.40-GHz Core i7 quad-core processor (Intel, Santa Clara, CA) with 8 GB memory. Except for the EMAN2 and ASPIRE class-averaging steps, the entire algorithm runs on a single core. In comparison, almost all other ab initio algorithms take multiple days, with only a single recent exception that is comparable to ours in terms of speed (13). Our algorithm was implemented in the software PYTHON (Python Software Foundation, python.org) with CYTHON extension (cython.org), and is available upon request. The second advantage is reducing the model complexity, i.e., reducing the possible three-dimensional structures that can be represented using our pseudo-atomic model. During the initial

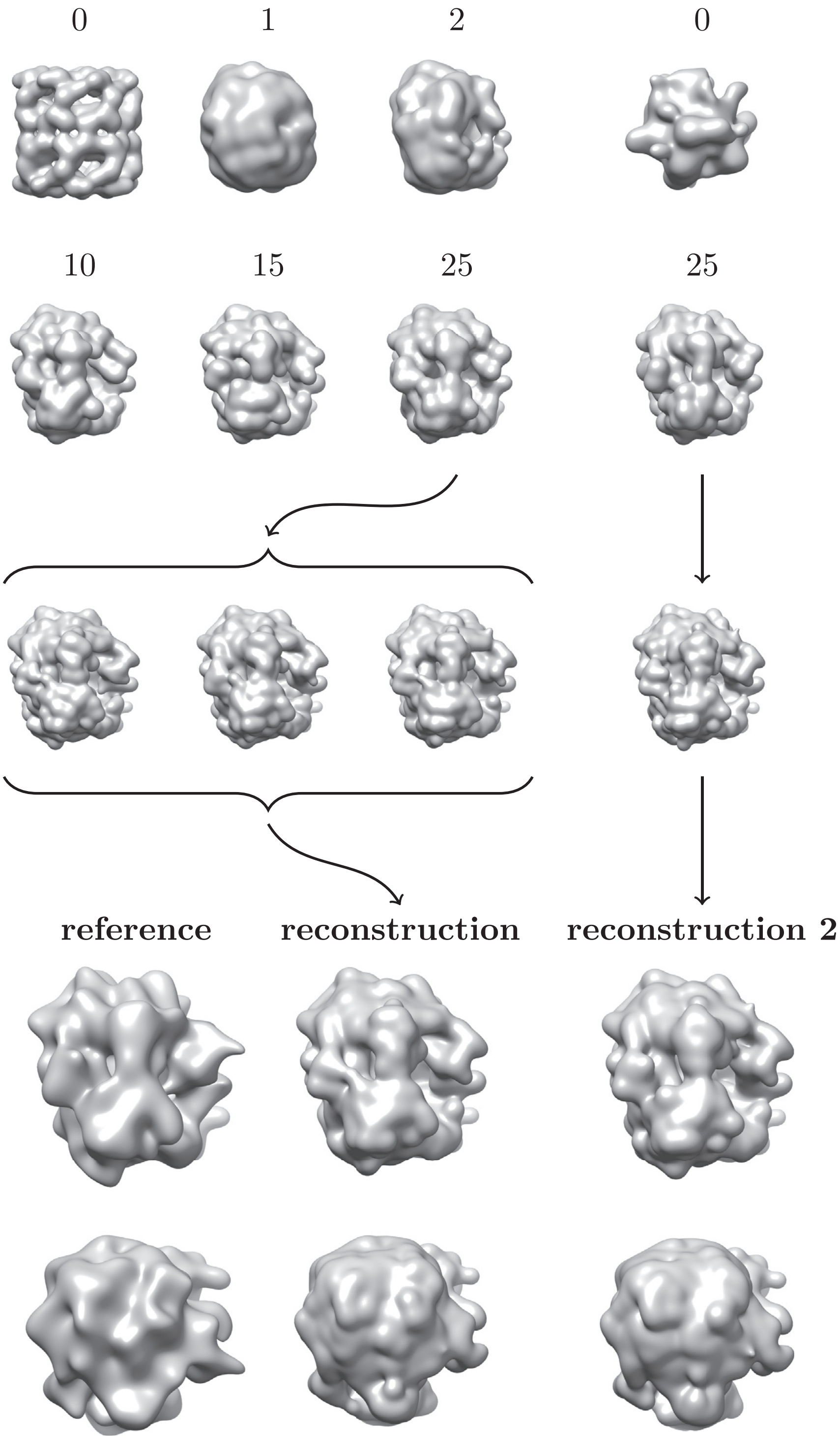

FIGURE 7 Results for the 70S ribosome, using real data. The algorithm was initialized with a model of an unrelated structure, GroEL, and successfully converged to the 70S structure. Shown below the labeled models from the initial stage are multiple models from the posterior distribution. These are averaged to obtain the final reconstruction (*last two rows*). We computed a second reconstruction starting from a different, random initial model (*rightmost column*). Once again the algorithm converged to the correct structure, showing its robustness to the choice of initial model. Shown as the reference is the reconstruction obtained by the PRIME algorithm, low-pass-filtered. The cross correlation between each of our reconstructions and the PRIME reconstruction is 0.900 for our first and 0.895 for our second reconstruction. The cross correlation between our reconstructions is 0.986.

stage of our algorithm when there are only a few large pseudo-atoms it is impossible to represent high-frequency information in the three-dimensional structure. Because we are interested in a low-resolution model, this excludes a large number of undesired models from our search space, thereby simplifying the problem and making the algorithm more robust. Some other reconstruction algorithms (both ab initio (12) and refinement (35)) apply a low-pass filter to the current volume at every iteration to achieve a similar effect. However, in our case, this is a property of the model itself.

Another principle difference to existing reconstruction methods is that we use Markov chain Monte Carlo sampling to generate an ensemble of models from the posterior

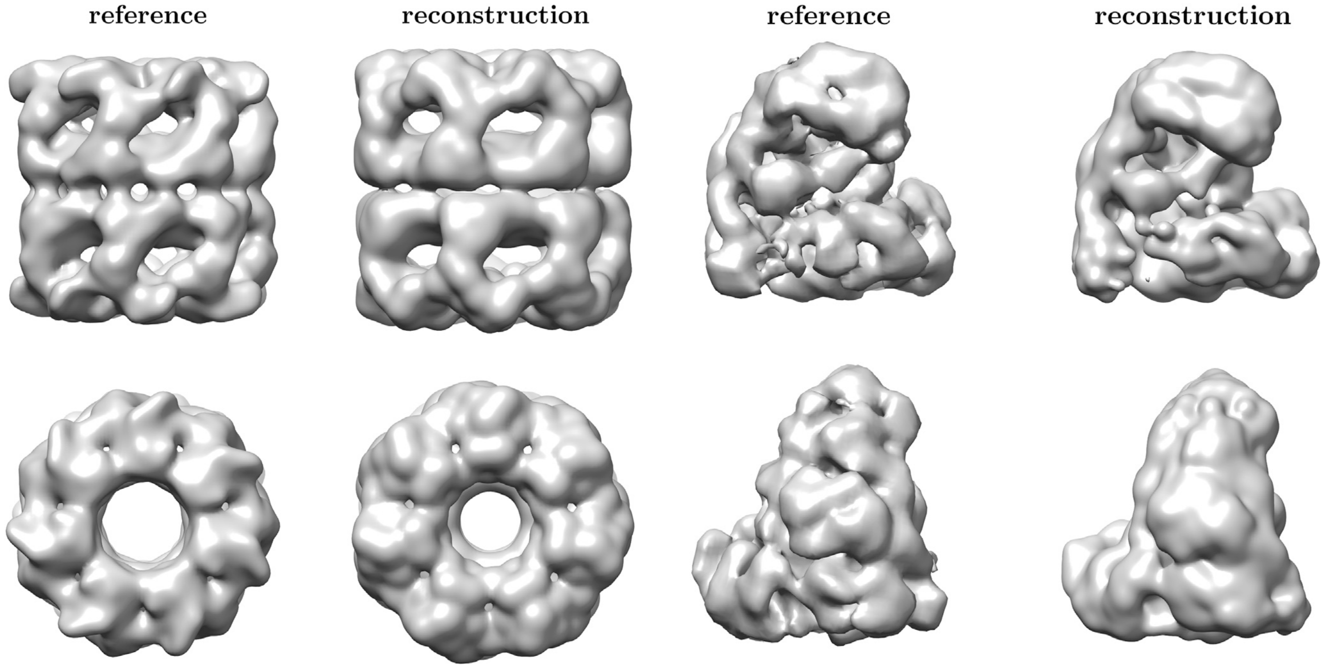

FIGURE 8 Results for GroEL, using real data. The final reconstruction agrees well with the reference at 20 Å, as shown by the cross-correlation value of 0.927.

distribution. This allows us to assess the ambiguity of the data when there are multiple reconstructions compatible with the projection images. It also adds to the robustness of our method. Moreover, we are able to estimate the precision of every model parameter including the pseudo-atom positions and the rotations (Figs. 5 and 9). Yet another advantage of sampling is that by computing the mean we can represent the final structure more accurately than would be possible using any single model, such as the MAP estimate. Compare, for example, any of the posterior models from the refinement stage with the final reconstruction in either Fig. 5 or 7. The individual pseudo-atoms are no longer visible in the posterior mean.

Using our pseudo-atomic model, it is possible to express many different forms of prior information about the structure, and the Bayesian framework dictates how to incorporate such prior information. In this article, nonnegativity and smoothness were used as prior information, by using nonnegative weights for the pseudo-atoms, and restricting all pseudo-atoms to be the same size. Another form of prior information that we demonstrated using GroEL is symmetry constraints, which can be imposed on the pseudo-atom positions for inferring initial models with known symmetry. Some extensions are straightforward, such as using a known low-resolution version of the structure as a prior distribution for the pseudo-atom positions, or using a nonuniform prior distribution for the rotations in the case of structures with preferred orientations. A more ambitious possibility for future work is to incorporate data from other sources, such as cross-linking/mass spectrometry, or crystallography. Another direction of future research is to modify the algorithm to handle conformational heterogeneity by inferring multiple structural conformations.

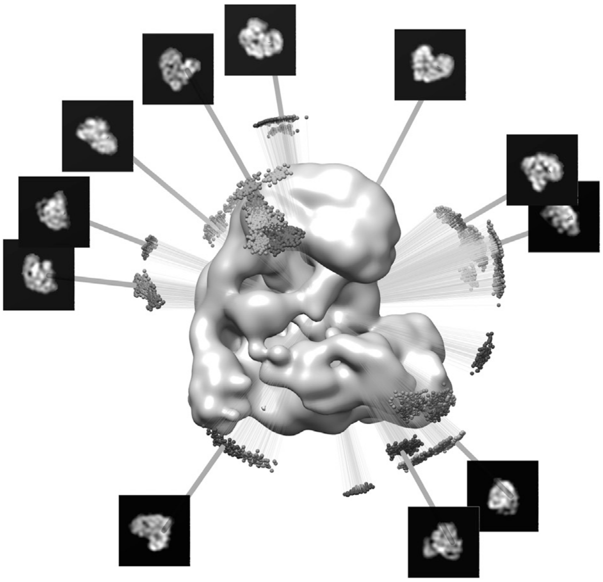

FIGURE 9 Results for experimental APC/C data. Class averages were computed from 10,000 raw particles in an ab initio setting, without making use of previous structures. The final reconstruction has a cross-correlation of 0.902 with the reference. At the bottom is the distribution of rotations at the end of the initial stage. Instead of estimating just a single rotation for each image, we obtain a cluster of rotations consistent with the image. The width of each cluster gives an indication of the precision of the estimated rotation.

SUPPORTING MATERIAL

Supporting Materials and Methods, three figures, and two movies are available at http://www.biophysj.org/biophysj/supplemental/S0006-3495(15)00064-8.

AUTHOR CONTRIBUTIONS

P.J. and M.H. designed and performed the research, and wrote the paper. P.J. contributed the analytic tools and analyzed the data.

ACKNOWLEDGMENTS

We thank Niels Fischer and Holger Stark (MPI for Biophysical Chemistry Göttingen) for providing us with the APC/C data. We also thank Hans Elmlund for providing us with the 70S ribosome initial model obtained using the software PRIME (12). Molecular graphics and analyses were performed with the UCSF CHIMERA software package (36).

This work was supported by Deutsche Forschungsgemeinschaft grants No. HA 5918/1-1 and No. SFB860 TP B9.

REFERENCES

1. Frank, J. 2006. Three-Dimensional Electron Microscopy of Macromolecular Assemblies. Oxford University Press, New York.
2. Penczek, P. A., R. A. Grassucci, and J. Frank. 1994. The ribosome at improved resolution: new techniques for merging and orientation refinement in 3D cryo-electron microscopy of biological particles. *Ultramicroscopy.* 53:251–270.
3. Scheres, S. H. W., M. Valle, …, J. M. Carazo. 2005. Maximum-likelihood multi-reference refinement for electron microscopy images. *J. Mol. Biol.* 348:139–149.
4. Penczek, P. A., J. Zhu, and J. Frank. 1996. A common-lines based method for determining orientations for $N > 3$ particle projections simultaneously. *Ultramicroscopy.* 63:205–218.
5. van Heel, M. 1987. Angular reconstitution: a posteriori assignment of projection directions for 3D reconstruction. *Ultramicroscopy.* 21:111–123.
6. Elmlund, H., J. Lundqvist, …, M. Lindahl. 2008. A new cryo-EM single-particle ab initio reconstruction method visualizes secondary structure elements in an ATP-fueled AAA+ motor. *J. Mol. Biol.* 375:934–947.
7. Singer, A., and Y. Shkolnisky. 2011. Three-dimensional structure determination from common lines in cryo-EM by eigenvectors and semidefinite programming. *SIAM J.* 4:543–572.
8. Elmlund, D., and H. Elmlund. 2012. SIMPLE: software for ab initio reconstruction of heterogeneous single-particles. *J. Struct. Biol.* 180:420–427.
9. Lyumkis, D., S. Vinterbo, …, B. Carragher. 2013. OPTIMOD—an automated approach for constructing and optimizing initial models for single-particle electron microscopy. *J. Struct. Biol.* 184:417–426.
10. Yan, X., K. A. Dryden, …, T. S. Baker. 2007. Ab initio random model method facilitates 3D reconstruction of icosahedral particles. *J. Struct. Biol.* 157:211–225.
11. Sanz-García, E., A. B. Stewart, and D. M. Belnap. 2010. The random-model method enables ab initio 3D reconstruction of asymmetric particles and determination of particle symmetry. *J. Struct. Biol.* 171:216–222.
12. Elmlund, H., D. Elmlund, and S. Bengio. 2013. PRIME: probabilistic initial 3D model generation for single-particle cryo-electron microscopy. *Structure.* 21:1299–1306.
13. Vargas, J., A. L. Alvarez-Cabrera, …, C. O. Sorzano. 2014. Efficient initial volume determination from electron microscopy images of single particles. *Bioinformatics.* 30:2891–2898.
14. Jaitly, N., M. A. Brubaker, …, R. H. Lilien. 2010. A Bayesian method for 3D macromolecular structure inference using class average images from single particle electron microscopy. *Bioinformatics.* 26:2406–2415.
15. Scheres, S. H. W., H. Gao, …, J. M. Carazo. 2007. Disentangling conformational states of macromolecules in 3D-EM through likelihood optimization. *Nat. Methods.* 4:27–29.
16. Scheres, S. H. W. 2010. Maximum-likelihood methods in cryo-EM. Part II: application to experimental data. *Methods Enzymol.* 482:295–320.
17. Scheres, S. H. W. 2012. A Bayesian view on cryo-EM structure determination. *J. Mol. Biol.* 415:406–418.
18. Scheres, S. H. W. 2012. RELION: implementation of a Bayesian approach to cryo-EM structure determination. *J. Struct. Biol.* 180:519–530.
19. Kawabata, T. 2008. Multiple subunit fitting into a low-resolution density map of a macromolecular complex using a Gaussian mixture model. *Biophys. J.* 95:4643–4658.
20. Nogales-Cadenas, R., S. Jonic, …, C. O. Sorzano. 2013. 3DEM LOUPE: analysis of macromolecular dynamics using structures from electron microscopy. *Nucleic Acids Res.* 41:W363–W367.
21. Jin, Q., C. O. S. Sorzano, …, S. Jonić. 2014. Iterative elastic 3D-to-2D alignment method using normal modes for studying structural dynamics of large macromolecular complexes. *Structure.* 22:496–506.
22. McLachlan, G., and D. Peel. 2000. Finite Mixture Models, Wiley Series in Probability and Statistics. John Wiley, New York.
23. Dempster, A. P., N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. A.* 39:1–38.
24. Geman, S., and D. Geman. 1984. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* 6:721–741.
25. Byrd, R. H., P. Lu, …, C. Zhu. 1995. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.* 16:1190–1208.
26. Habeck, M. 2009. Generation of three-dimensional random rotations in fitting and matching problems. *Comput. Stat.* 24:719–731.
27. Rosenthal, P. B., and R. Henderson. 2003. Optimal determination of particle orientation, absolute hand, and contrast loss in single-particle electron cryomicroscopy. *J. Mol. Biol.* 333:721–745.
28. Tang, G., L. Peng, …, S. J. Ludtke. 2007. EMAN2: an extensible image processing suite for electron microscopy. *J. Struct. Biol.* 157:38–46.
29. Frank, J. 2014. 70S *E. coli* ribosome. Protein Data Bank. http://www.ebi.ac.uk/pdbe/emdb/test_data.html. Accessed December 2014.
30. Zhao, Z., and A. Singer. 2014. Rotationally invariant image representation for viewing direction classification in cryo-EM. *J. Struct. Biol.* 186:153–166.
31. EMAN Wiki. 2014. EMAN2.1 Workshops, Summer 2014. http://blake.bcm.edu/emanwiki/Ws2014. Accessed December 2014.
32. Frye, J. J., N. G. Brown, …, B. A. Schulman. 2013. Electron microscopy structure of human APC/C(CDH1)-EMI1 reveals multimodal mechanism of E3 ligase shutdown. *Nat. Struct. Mol. Biol.* 20:827–835.
33. Marabini, R., G. T. Herman, and J. M. Carazo. 1998. 3D reconstruction in electron microscopy using ART with smooth spherically symmetric volume elements (blobs). *Ultramicroscopy.* 72:53–65.
34. Sorzano, C. O. S., R. Marabini, …, A. Pascual-Montano. 2004. XMIPP: a new generation of an open-source image processing package for electron microscopy. *J. Struct. Biol.* 148:194–204.
35. Scheres, S. H. W., and S. Chen. 2012. Prevention of overfitting in cryo-EM structure determination. *Nat. Methods.* 9:853–854.
36. Pettersen, E. F., T. D. Goddard, …, T. E. Ferrin. 2004. UCSF CHIMERA—a visualization system for exploratory research and analysis. *J. Comput. Chem.* 25:1605–1612.

Supplementary materials for *Bayesian inference of initial models in cryo-electron microscopy using pseudo-atoms*

Paul Joubert [1*] and Michael Habeck [1,2*]

[1] Felix-Bernstein Institute for Mathematical Statistics, Georg-August-Universität Göttingen
[2] Max Planck Institute for Biophysical Chemistry, Göttingen

1 FSC curves

Figure 1 shows the FSC curves between the references and the reconstructions.

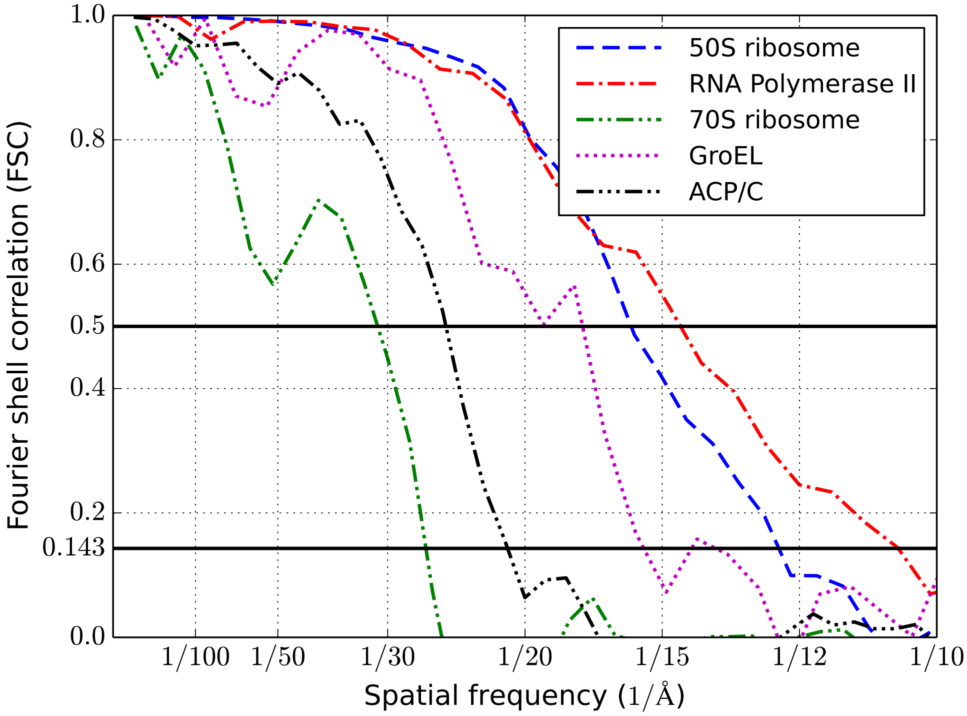

Figure 1: FSC curves comparing the references with the reconstructions. The references are the same as shown in the main text, i.e. for the 50S ribosome, RNA Polymerase II and GroEL they are created from the atomic structures at 25 Å, 20 Å and 20 Å respectively, for the 70S ribosome the reference was obtained by the PRIME algorithm and for the last structure (APC/C) the reference is from another publication (EMD-2354). The normalized cross-correlations for the same five pairs of structures are 0.990, 0.966, 0.900, 0.927 and 0.902 respectively.

[*] to whom correspondence should be addressed

2 Movies

The first movie (movie1.avi) shows multiple pseudo-atomic models of the same structure (RNA Polymerase II) with increasing numbers of pseudo atoms. It demonstrates that a small number of pseudo-atoms are sufficient for representing low-resolution structures, and that the number of model parameters required are orders of magnitude fewer than with the standard grid-based representation.

The second movie (movie2.avi) shows a reconstruction of the 50S ribosome, from the initial random model to the final model. The trajectories of the individual pseudo-atoms as well as those of the individual rotations can be seen. For the rotations, only the projection direction is shown (the first two Euler angles), not the in-plane rotation component (the third Euler angle).

3 Computing the posterior distribution

Here we give a more formal description of the different components of the Bayesian framework, starting with the data $\mathcal{D}$.

The i'th deconvolved class average is a non-negative grayscale image $(I_{ij})_{1\leq j\leq n}$ with 2D pixel coordinates[1] $(x^o_{ij})_{1\leq j\leq n}$, where the pixels are indexed by j, and n is the number of pixels per image. The I_{ij}'s for a given i are then multiplied by a constant scaling factor $\alpha \geq 0$, and rounded to the nearest integer to obtain $y_{ij} := \mathrm{round}(\alpha I_{ij})$. The constant α is chosen such that $\sum_j y_{ij} \approx C$ for a previously fixed constant C. The I_{ij}'s are discarded, and we continue with the y_{ij}'s.

The observed data is now considered as 2D points, with y_{ij} points at each pixel centered at x^o_{ij}. Every pixel gives rise to a y_{ij}-dimensional data vector

$$d_{ij} = [x^o_{ij1}, \ldots, x^o_{ijl}, \ldots, x^o_{ijy_{ij}}]$$

with identical entries $x^o_{ijl} = x^o_{ij}$, where l runs from 1 to y_{ij}. All the data vectors d_{ij} together form the observed data $\mathcal{D} = x^o = \{x^o_{ijl}\}$.

Before describing the forward model, we introduce the latent variables. These are the assignments z as typically used for Gaussian mixture models, and the missing z-components x^m.

The assignments $z = \{z_{ijl}\}$ consist of one assignment z_{ijl} for each point x_{ijl}, indicating the mixture component responsible for generating the point. We use 1-of-K notation, whereby z_{ijl} is a length K vector $(z_{ijlk})_k$ with $z_{ijlk} \in \{0,1\}$ and $\sum_k z_{ijlk} = 1$. I.e. the k for which $z_{ijlk} = 1$ indicates the component that generated x_{ijl}.

The missing components $x^m = \{x^m_{ijl}\}$ consist of the z-component x^m_{ijl} for each sampled 3D point $x_{ijl} = [x^o_{ijl}\ x^m_{ijl}]$. Since we only observe the first two coordinates, i.e. x^o_{ijl}, the z-component is referred to as missing.

We write $\mathcal{Z} = \{z, x^m\}$ for all latent variables together.

Given a model parameterized as described above, the observed data can be generated as follows: for a given direction i compute the 2D density $I_i(x)$. Sample C points from this density, and create a 2D histogram with bins centered at the grid points x^o_{ij}. Then y_{ij} is defined as the number of points in the j'th bin. If the grid is sufficiently fine, we can make the following assumption to simplify the forward model: all the points in the j'th bin are replaced by the center of the bin, x^o_{ij}. In other words we assume that we sampled y_{ij} copies of x^o_{ij}, for each j.

This forward model can also be described in a slightly different way which will be used below in formulating the sampling algorithm. Instead of first projecting the 3D density to 2D, and then sampling C points, we could equivalently first sample C 3D points, and then project them to 2D. The 3D points sampled from the rotated density are denoted by $x_{ijl} = [x^o_{ijl}\ x^m_{ijl}]$, where $x^o_{ijl} \in \mathbb{R}^2$, and $x^m_{ijl} \in \mathbb{R}$. Projection along the z-axis means discarding the z-component x^m_{ijl}, i.e. x_{ijl} is projected to x^o_{ijl}.

[1] The superscripts o and m stand for *observed* and *missing* respectively, as will be explained shortly.

The above forward model describes the extended likelihood for the data and latent variables:

$$p(\mathcal{D}, \mathcal{Z}|\theta) = p(x^o, x^m, z|\mu, s, w, R, t) \tag{1}$$

$$= \prod_{ijlk} w_k^{z_{ijlk}} f(x_{ijl}|R_i\mu_k + t_i|\frac{1}{s}I)^{z_{ijlk}}. \tag{2}$$

Marginalizing out the latent variables gives the data likelihood:

$$p(\mathcal{D}|\theta) = p(x^o|\mu, s, w, R, t) \tag{3}$$

$$= \iint p(x^o, x^m, z|\mu, s, w, R, t)\mathrm{d}x^m\mathrm{d}z \tag{4}$$

$$= \int p(\mathcal{D}, \mathcal{Z}|\theta)\mathrm{d}\mathcal{Z}. \tag{5}$$

The prior is assumed to factorize over the model parameters:

$$p(\theta) = p(\mu, s, w, R, t) = p(\mu)p(s)p(w)p(R)p(t), \tag{6}$$

where the means $p(\mu) = \prod_k f(\mu_k|0, \frac{1}{r}I)$ follow normal distributions, the precision $p(s) \propto s^{a-1}e^{-bs}$ a gamma distribution, the weights $p(w) \propto \prod_k w_k^{\lambda-1}$ a Dirichlet distribution, the rotations are distributed uniformly, and the translations $p(t) = \prod_i f(t_i|0, \frac{1}{r}I)$ follow normal distributions. The hyperparameters r, a, b and λ are kept fixed.

4 Gibbs sampling

Here we give the equations for performing Gibbs sampling. These are uniquely determined given the forward model and prior defined above. We use Gibbs sampling to sample from the extended posterior $p(\mathcal{Z}, \theta|\mathcal{D})$, and then discard the latent variables $\mathcal{Z}$ to obtain samples from the posterior. The extended posterior is proportional to the product of the extended likelihood (Equation 2) and the prior (Equation 6). To sample from this extended posterior using Gibbs sampling we compute the conditional distribution for each of the parameters, conditioned on all the other parameters. They are all standard distributions (Gaussian, multinomial, Dirichlet and gamma) except for the rotations, which are of the form $\exp[\mathrm{tr}(A^TR)]$.

The conditional for each assignment is a multinomial distribution:

$$p(z_{ijl}|x^o_{ijl}, w, \mu, s, R, t) = \prod_k w_k^{z_{ijlk}} f(x^o_{ijl}|P(R_i\mu_k + t_i), \frac{1}{s}I)^{z_{ijlk}}.$$

where

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

The conditional for the missing z-component for a single point is a 1D normal distribution:

$$p(x^m_{ijl}|x^o_{ijl}, z_{ijl}, \mu, s, R, t) = \prod_k f(x^m_{ijl}|P_z(R_i\mu_k + t_i), \frac{1}{s})^{z_{ijlk}},$$

where

$$P_z = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}.$$

The conditional for the weights is a Dirichlet distribution:

$$p(w|x^m, z, \mu, s, R, t) \propto \prod_k w_k^{n_k+\lambda-1}$$

where

$$n_k = \sum_{ijl} z_{ijlk}.$$

The conditional for each mean is a normal distribution:

$$p(\mu_k | x^m, z, s, w, R, t, x^o) = f(\mu_k | \mu, \Sigma)$$

where

$$\begin{aligned}\mu &= \frac{s}{sn_k + r} \sum_{ijl} z_{ijlk} R_i^T (x_{ijl} - t_i) \\ \Sigma &= \frac{1}{sn_k + r} I.\end{aligned}$$

The conditional for the precision is a gamma distribution:

$$p(s | x^m, z, \mu, R, t) \propto s^{a'-1} e^{-b's},$$

where

$$\begin{aligned}a' &= a + \frac{3}{2} N \\ b' &= b + \frac{1}{2} \sum_{ijkl} z_{ijkl} \left[\| x_{ijl}^o - P(R_i \mu_k + t_i) \|^2 + (x_{ijl}^m - P_z(R_i + t_i)\mu_k)^2 \right].\end{aligned}$$

The conditional for each rotation is

$$p(R_i | t_i, x^m, \mu, s, x^o) \propto \exp \left[\mathrm{tr}(A_i^T R_i) \right],$$

where

$$A_i = s \sum_{jlk} z_{ijlk} (x_{ijl} - t_i) \mu_k^T.$$

The conditional for each translation is a normal distribution:

$$p(t_i | x, \mu, s, R) = f(t_i | \mu, \Sigma)$$

where

$$\begin{aligned}\mu &= \frac{\sum_{jlk} z_{ijlk} (x_{ijl} - R_i \mu_k)}{\sum_{jlk} z_{ijlk}} \\ \Sigma &= \frac{1}{s \sum_{jlk} z_{ijlk}} I.\end{aligned}$$

5 Prior hyperparameters

The prior distribution on the pseudo-atom size σ is given by a gamma distribution over the precision $s = 1/\sigma^2$:

$$p(s) = \frac{\beta^\alpha}{\Gamma(\alpha)} s^{\alpha-1} e^{-\beta s}.$$

The mean α/β and variance α/β^2 of this distribution encodes our prior knowledge about the size of the pseudo-atoms. In Figure 2 on the left are a few examples for different values of α and β, with β chosen such that the mean is $1/10^2$ (i.e. $\beta = 10^2\alpha$). In the same figure on the right are the effects of the different choices of the hyperparameters on a reconstruction of the 50S ribosome from simulated data. It shows that

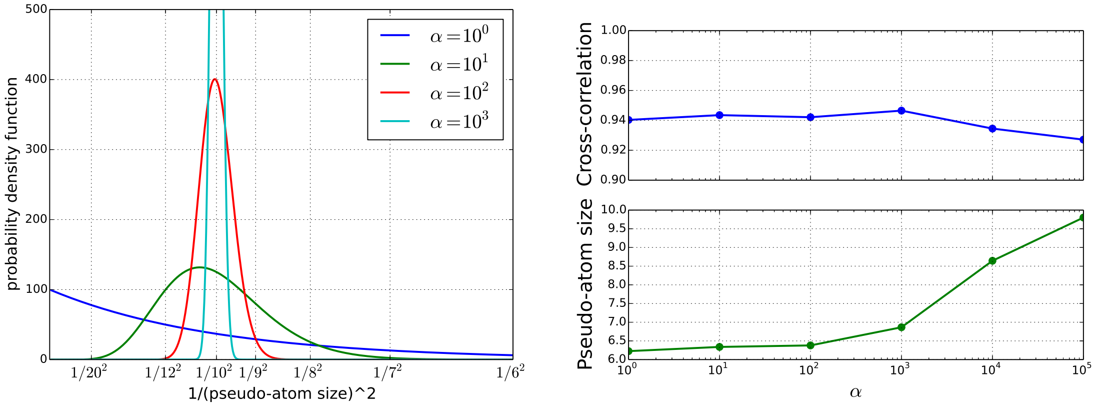

Figure 2: Varying the prior on the pseudo-atom size. On the left are different prior distributions over the pseudo-atom precision s, which is related to the pseudo-atom size σ by $s = 1/\sigma^2$. They correspond to a wide range of different choices of the hyperparameter α. The other hyperparameter β determining the prior distribution is chosen to ensure that the mean of s is $1/10^2$. Low values of α place effectively no restriction on the pseudo-atom size, while very high values of α restrict the pseudo-atom size to be very close to 10 Å. This can be seen on the right, where a 50S ribosome model inferred from simulated data is compared to a reference model. When a broad prior on the pseudo-atom size is used (i.e. α is low), then then final size is around 6 Å. As the prior becomes narrower (i.e. α is high), the final size tends to the mean prior value of 10 Å. The figure shows that for α in the range 1 to about 1000, the quality of the result as measured by the cross-correlation does not depend on the specific choice of α.

good results are obtained for all values of the hyperparameters, although the results deteriorate for very high values of α (above 1000). We conclude that the specific value of alpha is not very important for our algorithm, and recommend it to be chosen in the range 1 to 1000. We used $\alpha = 10$ for our experiments. The value of β can be chosen as was done here to ensure that the mean is 10, although a similar experiment shows that the exact value of β is also not very important. For our experiments we used $\beta = 10^2\alpha = 1000$.

The prior distribution on the pseudo-atom weights $w = \{w_k\}$ is a Dirichlet distribution

$$p(w) \propto \sum_{k=1}^{K} w_k^{\lambda-1}.$$

This distribution is parametrized by a single hyperparameter λ, which determines the allowable variation among the weights for the different pseudo-atoms. Higher values of λ lead to less variation.

In Figure 3 we show the effect of varying λ on the quality of a reconstruction using the same data as before. The figure shows that all values of λ in a wide range lead to similar results, although very small values of λ (0.1 and 1.0) give slightly worse results. We therefore recommend choosing λ in the range from 10 to 10^5. We used the value $\lambda = 1000$ for our experiments.

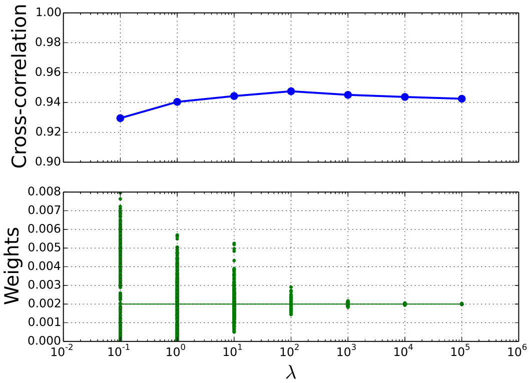

Figure 3: Varying the prior on the pseudo-atom weights. Multiple reconstructions using the same 50S ribosome data as before were performed using a range of values of λ, the hyperparameter for the weights. At the top are the resulting cross-correlations with the reference, measuring the quality of the inferred models. The cross-correlations are slightly lower for very low values of λ, but stay relatively constant for λ above 10. At the bottom are the individual pseudo-atom weights, showing that the variation in the weights decreases with increasing λ.