AlignOT: An Optimal Transport Based Algorithm for Fast 3D Alignment With Applications to Cryogenic Electron Microscopy Density Maps

Aryan Tajmir Riahi, Geoffrey Woollard, Frédéric Poitevin, Anne Condon, and Khanh Dao Duc

Abstract—Aligning electron density maps from Cryogenic electron microscopy (cryo-EM) is a first key step for studying multiple conformations of a biomolecule. As this step remains costly and challenging, with standard alignment tools being potentially stuck in local minima, we propose here a new procedure, called AlignOT, which relies on the use of computational optimal transport (OT) to align EM maps in 3D space. By embedding a fast estimation of OT maps within a stochastic gradient descent algorithm, our method searches for a rotation that minimizes the Wasserstein distance between two maps, represented as point clouds. We quantify the impact of various parameters on the precision and accuracy of the alignment, and show that AlignOT can outperform the standard local alignment methods, with an increased range of rotation angles leading to proper alignment. We further benchmark AlignOT on various pairs of experimental maps, which account for different types of conformational heterogeneities and geometric properties. As our experiments show good performance, we anticipate that our method can be broadly applied to align 3D EM maps.

Index Terms—3D alignment, electron microscopy, image registration, optimal transport, protein.

I. INTRODUCTION

SOLVING the 3D structures of biomolecules is key to their function and the mechanisms underlying biological processes. For this purpose, cryogenic electron microscopy (cryo-EM) has become in recent years the most used technique to solve structures [1]. One main advantage of this technique, in contrast with X-ray crystallography, is that it potentially allows various conformations (or 3D configurations) of the same molecule to be solved [2]. Once different conformations are obtained as 3D EM density maps (i.e., large 3D grids of voxels with different levels of intensities), aligning these maps in 3D space is needed to further compare them. More generally, solving the rigid body alignment problem for density maps and/or atomic models has applications in various other tasks, such as model building [3] and subtomogram averaging [4].

Efficient methods have been developed to align two protein structures [5], [6], assuming their atomic composition is known. In this case, aligning two conformational structures is tantamount to finding an optimal rigid body transformation (i.e., a combination of 3D translation and rotation) that can align homologous atoms. However, when density maps are only given, one cannot directly establish such a homology correspondence from voxel to voxel, so the same problem becomes more challenging as the grid size increases and with the computational cost of searching over all possible rigid body transformations.

To solve the rigid body alignment problem for 3D cryo-EM density maps, standard approaches use various algorithms to maximize correlation [7], [8], [9] or minimize root mean squared displacement (RMSD) [10]. More recently, Han et al. introduced a new method, which relies on representing the maps as sets of unit vectors before performing alignment [11]. Overall, both the choice of the metric to optimize, as well as the representation of the maps, can play important roles in getting a successful alignment. In this paper, we introduce a novel approach, called AlignOT, that uses a point-cloud representation of 3D maps, and minimizes the so-called Wasserstein distance between two maps with a stochastic gradient algorithm. This non-euclidean distance is associated with the theory of Optimal Transport (OT) [12], with recent advances that make tractable the computation of transport-based distances [13], [14]. After describing the procedure of AlignOT in detail, we present the results of our experiments that quantify the precision and accuracy of this method, and benchmark it on a set of representative experimental maps. Overall, the good performance of AlignOT with respect to standard local alignment methods suggests that our method can be broadly applied, as an alternative for aligning 3D EM maps with a non-euclidean metric. We finally discuss the potential limitations of AlignOT, and connections with other recent methods and problems in optimal transport that would help to further improve and generalize it.
II. MATERIAL AND METHODS

A. Point Cloud Representation of EM Maps

To represent voxelized cryo-EM maps, we follow the same procedure as in Wriggers et al. [15], that uses the topology representing network algorithm (TRN) [16] to reduce a map of \(d^3\) voxels to a point cloud, i.e., a set of \(n\) points \(\in \mathbb{R}^3\). Briefly, the voxelized maps (typically \(\sim 100^3\) to \(\sim 500^3\) voxels in the 3D grid) are first thresholded with a noise floor, to set low intensity regions to zero, and normalized to a probability mass function over the grid points. \(n\) points \((r_i(0))_{i=1...n}\) are initially sampled from \(\mathcal{P}\), and then updated in parallel for \(t_f\) rounds by taking weighted steps, according to the following equations [16]

\[
\begin{align*}
\mathbf{r}_i(t + 1) &= \mathbf{r}_i(t) + \varepsilon(t) \exp[-k_i/\lambda(t)](\mathbf{r}_i - \mathbf{r}_i(t)), \\
\lambda_i(t) &= \lambda_0 \left(\frac{\lambda_f}{\lambda_0}\right)^{t/t_f}, \\
\varepsilon(t) &= \varepsilon_0 \left(\frac{\varepsilon_f}{\varepsilon_0}\right)^{t/t_f},
\end{align*}
\]

where \(r_i\) is a single grid point sampled from \(\mathcal{P}\), \(k_i\) denotes the rank of point \(r_i\), by its euclidean distance to \(r_1\), and \(\lambda_0, \varepsilon_0, \lambda_f, \varepsilon_f, t_f\) are hyperparameters. In practice, we used \(\varepsilon_0 = 0.3\) and \(\varepsilon_f = 0.05\) for the initial and final step sizes, \(\lambda_0 = 0.005 \times n\) and \(\lambda_f = 0.5\) for the initial and final scaling ranks, and \(t_f = 8 \times n\) for the total number of steps (adapted from [17]).

B. Optimal Transport and Wasserstein Distance

To compare the point cloud representations of EM maps, we use a non-euclidean metric that derives from the theory of Optimal Transport [12]. For two given point clouds, \(A = \{a_1,...,a_n\}\) and \(B = \{b_1,...,b_n\}\), we define a cost matrix \(C_{i,j} = d(a_i, b_j)^2\), where \(d\) is the euclidean distance. The entropy regularized 2-Wasserstein distance between \(A\) and \(B\), denoted by \(W_{2,\varepsilon}(A, B)\), is then defined as

\[
W_{2,\varepsilon}(A, B) = \left[\min_{P \in \mathbb{R}^{n \times n}} \sum_{i,j=1}^{n} C_{i,j}P_{i,j} + \varepsilon H(P)\right]^{1/2}
\]

s.t. \(P,\|P\| = P^T,\|P\| = 1/n\),

where \(\varepsilon \in \mathbb{R}_+\) is the regularization parameter and the entropy \(H(P)\) is given by

\[
H(P) = \sum_{i,j=1}^{n} P_{i,j} \log P_{i,j}.
\]

The minimizer of (4) is called the transport plan. For the rest of the Methods section, we will simply denote the Wasserstein distance as \(W_{2,\varepsilon}\), and \(P_{a(i),b(j)}\) as \(P_{a,b}\), where \(i(a)\) and \(i(b)\) are the indices of the two points \(a\) and \(b\) in \(A\) and \(B\), respectively.

C. AlignOT: Algorithm for 3D Map Alignment

To align two 3D maps from their point cloud representations \(A\) and \(B\), we solve the optimization problem

\[
q_{opt} = \arg \min_{q \in H} W_{2,\varepsilon}(R_q(A), B),
\]

where \(q\) is a quaternion (defined over the quaternion space \(\mathbb{H}\)), that we identify to a 3D rotation \(R_q\) in \(SO(3)\), so \(R_q(A) = \{R_q(a_j)\}_{a_j \in A}\). We explain in the Results Section III-A and Appendix A (SI File), available online, why we can only consider rotations and ignore translations to solve the general alignment problem, and provide more details on the identification of \(q\) to \(R_q\) in Appendix B (SI File), available online. Our stochastic gradient descent procedure to solve (6), called AlignOT, is detailed in Algorithm 1. At each iteration, the algorithm updates \(q\) from the transport plan \(P\) between \(R_q(A)\) and \(B\) as follows: After sampling one point \(a \in R_q(A)\), we evaluate \(\pi(a) = \arg \max_{b \in B} P_{a,b}\), and compute the gradient in \(q\) associated with \(d(\pi(a), a)^2\), where \(d\) is the euclidean distance, to update \(q\). We derive the analytical formula used to compute this gradient in Appendix B, available online. To compute the transport plan, we apply the Sinkhorn algorithm [13], with the initial vectors set as the outputs of the previous iteration to improve the convergence speed. In practice, we also set the convergence condition \(|d(\pi(a), a)|^2 < \delta\) (where \(\delta > 0\), that stops the algorithm before the maximum number of iterations. The hyperparameters of this procedure are the learning rate \(\alpha\) associated with gradient descent, the regularization parameter \(\epsilon\) associated with the Wasserstein distance, and a threshold \(\delta\) associated with the number of iterations. In all our experiments, we set \(\epsilon = 100\), \(\delta = 10^{-10}\), and the maximum number of iterations equal to 500. We used an adaptive approach for the learning rate (ADAGRAD [18]) and Bayesian Optimization [19, 20] to set the learning rate parameter \(\alpha\) to 0.1.

D. Implementation

We implemented AlignOT in Python 3.6.4. To sample a point cloud representation of an EM map using TRN, we adapted code from ProDy [21]. We used the NumPy package for matrix operations and POT’s implementation of the Sinkhorn algorithm, which was modified to set the initial vectors (instead of initializing with uniform vectors). Our code, which has options for aligning maps with both local or global search, is available in this GitHub repository.

E. Datasets

We tested AlignOT on various publicly available EM maps available from the EMDB, as listed in Table I, and shown in Fig. 1. For experiments that involved aligning a pair of distinct maps, we also used the corresponding pair of structures taken from the PDB [22], and used MM-align [5] to set a ground truth alignment, before converting the structures into EM maps using the function molmap in Chimera X [23]. All the datasets used in this study are available at this OSF page.
EM MAPS USED IN OUR EXPERIMENTS, ALSO SHOWN IN FIG. 1

| EMDB ID  | Protein or Complex Name | Figure |
|----------|-------------------------|--------|
| 1717     | Ribosome                | a      |
| EMDB/PDB ID | Protein or Complex Name | Figure/ID |
| 3240 / 5fn5 [24] | Human γ-secretase | 1 b / 1 |
| 2677 / 7a53 [25] | TRPML | 1 c / 2 |
| 8881 / 5wpq [27] | Voltage-gated calcium channel | 1 d / 3 |
| 9515 / 5gjw [29] | Yeast V-ATPase | 1 e / 4 |
| 6427 / 3jb [30] | Hsp90-Cdc37-Cdk4 | 1 f / 5 |
| 5284 / 3pfr [31] | eIF2B-eIF2 | 1 g / 6 |

The map from EMDB:1717 is used for aligning two copies of the same map. The other maps are grouped and aligned in our experiments as pair of different conformations of the same molecule, with the PDB structures used to obtain a ground truth (see Datasets and Results sections).

![Fig. 1](image-url) 3D maps used in our experiments, visualized with Chimera [7]. (a): Density map used in our first experiment (EMDB:1717). (b-g): Pairs of maps used in the second set of experiments, representing different conformational states of a given complex/molecule (see also Table 1).

III. RESULTS

A. Formulation of the Alignment Problem for EM Maps With Point Cloud Representation

We present here a new procedure, called AlignOT, that aligns 3D density maps from cryo-EM. Assuming that the EM maps are represented by two 3D point clouds \( A = \{a_1, \ldots, a_n\} \) and \( B = \{b_1, \ldots, b_n\} \), and for a given distance function \( d \) defined over the space of point clouds, the problem of aligning these maps consists of finding a rigid body transformation that minimizes the objective function

\[
L_d(R, T) = d(\text{move}_{R,T}(A), B),
\]

where \( \text{move}_{R,T}(A) = \{Ra_i + T|a_i \in A\} \).

As the choice of \( d \) influences both the accuracy and the computational cost of the solution to the rigid body alignment problem, we here use the 2-Wasserstein distance, associated with the theory of Optimal Transport [12]. This distance can be used to compute distances between probability distributions, and is applied here more specifically for two distributions of 3D point clouds of same size. To efficiently evaluate this distance, we consider a regularized version (see (4) in the Methods Section II-B), denoted \( W_{2,\epsilon} \). Besides, given the centers of mass \( \bar{a} = \frac{1}{n} \sum_{i=1}^{n} a_i \) and \( \bar{b} = \frac{1}{n} \sum_{j=1}^{n} b_j \), and the centered point clouds \( A_c = \{a_{c_i} = a_i - \bar{a}|a_i \in A\} \) and \( B_c = \{b_{c_i} = b_i - \bar{b}|b_i \in B\} \), we can show that the optimal translation of the objective function (7) is

\[
T_{opt} = \bar{b} - R_{opt}\bar{a},
\]

Thus, the search for an optimal rigid body transformation in (10) can be simplified to rotations after matching the centers of mass of \( A, B \), leading to (6) of the Methods Section II-B. We provide a detailed proof of this result in Appendix A (SI File), available online.
B. AlignOT Optimization Procedure and Complexity

To search for an optimal rotation that minimizes the Wasserstein distance between two 3D point clouds, we use an iterative Stochastic Gradient Descent algorithm, detailed in Algorithm 1. Basically, the algorithm aims to improve the alignment at each iteration, by rotating a map according to an assignment provided by the evaluation of the Wasserstein distance, with quaternions used to represent the 3D rotations (see Appendix B (SI File), available online, for a full description). Since EM maps are defined on a 3D voxelized grid, we also first convert the maps into point clouds, using the TRN algorithm [16]. We cover the application of the TRN application, the Wasserstein distance and the optimization procedure, as well as its implementation in the Material and Methods Sections II-A, II-B, II-C, and II-D.

The complexity of our method (summarized in Algorithm 1) can be evaluated as follows: Assume that $L$ is the maximum number of iterations, $n$ the size of the point cloud, and $\epsilon$ the regularization parameter of the $W_{2,2}$ distance. Each iteration of this algorithm consists of three steps. First, rotating one point cloud, second, computing the OT plan matrix, and last, sampling a random point and computing the gradient. Rotating a point cloud requires computing the coordinates of each point after rotation and takes $O(n)$ time, where $n$ represents the number of points. To compute the OT plan matrix we use the Sinkhorn algorithm [14] which solves this problem in $O(n^2 \log ne^{-3})$ time, where $\epsilon$ is the regularization parameter. Finally, the gradient step takes $O(1)$ time. Overall, the most time-consuming part of this algorithm is computing the OT plan matrix at each iteration. Therefore, the overall time complexity of the algorithm is $O(n^2 L \log ne^{-3})$.}

C. Alignment Between Maps can be Obtained by Minimizing the Wasserstein Distance

To evaluate our method, we first tested AlignOT on aligning two point clouds that differ by a rotation only. To do so, we used an experimental map of a ribosome from EMDB 1717, shown in Fig. 1(a) (ribosome structures are broadly studied in cryo-EM [2], [35]). First, we sampled two clouds of 500 points, and applied a rotation defined in its axis-angle representation by an arbitrary axis, and an angle $\theta = 20^\circ$. Fig. 2(a) illustrates how the moving point cloud gets closer to the targeted one over the iterations of the algorithm, until the convergence criterion is reached. To confirm this visual impression, we repeated the procedure with different initial angles $\theta \in \{10^\circ, 30^\circ, 50^\circ, 70^\circ\}$. The corresponding Wasserstein distance obtained across the iterations is shown in Fig. 2(b), with all the four trajectories converging to the same value and resulting in a successful alignment. However, we also observed that as $\theta$ increases, it takes more iterations for the algorithm to converge, with longer periods of slow variations at the beginning of the procedure, suggesting that this alignment can only be achieved within a certain range of $\theta$. Aside from the initial angle, we also studied in Fig. 2(c) how the size of the point cloud can affect the convergence plot. For $\theta = 50^\circ$, decreasing the point cloud size from 500 to 250 leads to a potential loss of precision (with the Wasserstein distance between two aligned point clouds increasing from $\sim 115$ to 130), but with faster convergence from the algorithm (from $\sim 200$ to 50 iterations to converge), indicating some trade-off between accuracy and speed.

To interpret these results, we further plotted in Fig. 2(d) how the Wasserstein distance varies on average (after sampling
different point clouds of same size 250 and 500), as a function of $\theta$ (and same rotation axis). In addition to the global minimum achieved for $\theta = 0$, another local minimum was detected at $\theta = 180^\circ$, separated by a peak around $90^\circ$. While the sharp decrease observed towards the global minimum suggests that the optimization procedure can converge well for $\theta \leq 60^\circ$, it is also possible that the algorithm does not converge to the global minimum above this range. Besides, the higher variability obtained from sampling point clouds of size 250, compared with 500, suggests that the final alignment may be less accurate in this case. While the same fixed rotation axis was considered in the previous experiments (with different values of $\theta$), we finally evaluated the probability to successfully align the maps for initial rotations of fixed angle $\theta$ (45, 60, 75 and 90$^\circ$), and across different axes covering half of the sphere $S^2$. Upon mapping the axes on the planar disk in Fig. 2(e), we found local regions of poorer alignment that match with local minima of the Wasserstein distance. These results also confirm the existence of a limiting range within which the method can align two maps. While a successful alignment is overall obtained for $\theta = 45^\circ$, the maps get partially aligned in different regions of the disks for $60^\circ$ (see Fig. 2(e)), with the performance worsening as $\theta$ increases (see Supplementary Fig. S1, SI file, available online).

### D. Benchmarking Performance and Accuracy of AlignOT With a Pair of Identical Maps

We next focused on point cloud size and the range of convergence, and quantified their impact on the accuracy and computational cost of AlignOT. We first considered the same map and alignment task with $\theta = 20^\circ$ as previously, with different point cloud sizes $n$ (from 50 to 1000). The results obtained with a standard workstation are shown in Table II, and confirm the existence of a trade-off between accuracy and speed, with the alignment improving as $n$ increases, but with a larger runtime that grows from a few seconds for $n = 50$, 100, 200 to approximately a minute for $n = 1000$. While the accuracy of the alignment remains poor for $n = 50$ and 100 (with an average error of 12.6° and 7.72° respectively), it significantly improves for $n = 500$ with 2.35 ± 1.23° error observed, with a runtime suggesting that AlignOT can be used in practice on standard density maps. We also noted that AlignOT runs slower in comparison with Chimera and Chimera X’s alignment function fitmap, which performs a steepest ascent optimization to align maps according to their overlapping score [7], [23] (0.3 s on average). On the other hand, we show in our next experiments that AlignOT outperforms fitmap in accuracy as $\theta$ increases.

More precisely, we determined the range of $\theta$ within which the method converges, and compared our method with fitmap. As shown in Fig. 3, we found that AlignOT can cover a wider range that extends to 75°, while fitmap starts failing at approximately 45° (see also Supplementary Fig. S2 (SI File), available online, for a visualization of some representative alignments obtained). Beyond this value, AlignOT leads to some variable results up until 100°, due to the stochasticity of the algorithm. It then converges towards the other local minimum found for the Wasserstein distance observed in Fig. 2(d), with a difference of ~175° from the true alignment. In practice, both Chimera and our implementation of AlignOT provide a global search option that randomly generates different initial instances of the rotated map, and keeps the best alignment to the target map using fitmap. To estimate the potential gain from using AlignOT, we generated 500 random initial placements, and computed the rates of successful alignments from AlignOT and fitmap’s local search, to be 15.6% and 2.2%, respectively. In this context, AlignOT thus reduces the number of initial placements needed in the global search by a factor of 7.6 (as $(1 - 0.022)^7.6 \approx 1 - 0.156$).

### E. Benchmarking AlignOT With Heterogeneous Pairs

We finally tested AlignOT on pairs of distinct maps, as this reflects how the method should be applied in practice. More precisely, we considered six pairs, listed in Table I and shown in Fig. 1. Note that these pairs were recently used to evaluate another alignment method [11], and account for different conformations of a protein or complex structure (pairs b-g), as well

---

**TABLE II**

| $n$  | Angle difference | Runtime (s) |
|------|------------------|-------------|
| 50   | 12.60°±4.51°     | 2.36±0.42  |
| 100  | 7.72°±3.84°      | 4.66±0.57  |
| 200  | 3.85°±1.53°      | 4.14±0.57  |
| 500  | 2.35°±1.23°      | 16.36±1.37 |
| 1000 | 2.20°±0.96°      | 54.75±2.77 |

For point clouds of size $n$ (50, 100, 200, 500, 1000), and for a fixed rotation of $\theta = 20^\circ$, we ran AlignOT on EMDB:1717 50 times. We evaluate the accuracy of the method by computing the angle difference with the ground truth, and the associated runtime in seconds (mean and std). Experiments were run on an Intel(R) Core(TM) workstation with i7-7700HQ CPU @ 2.80GHz 2.81 GHz with 16.0 GB RAM.

---

Fig. 3. Comparison between AlignOT and Chimera’s fitmap function: With a fixed axis and for a rotation angle difference $\theta$ between 0 and 180 degrees ($x$-axis), we set up a couple of maps from EMDB:1717 and run AlignOT and Chimera’s fitmap function local search. The blue dots represent the error obtained in the alignment using fitmap, while the box plot (showing the minimum, first quartile, median, third quartile, and maximum) indicates the error of AlignOT over 10 runs.
Fig. 4. Performance of AlignOT with conformationally heterogeneous pairs (a): To quantify the performance of AlignOT on aligning different maps, we first aligned the corresponding PDB structures using MM-align [5] to set a ground truth alignment, and converted the aligned structures into cryo-EM density maps using the molmap command in Chimera [23], which were then used to benchmark Chimera’s fitmap function and AlignOT with different initial rotations (see also Table III). (b): Alignments obtained for the pair ID 4 (see Table I) and initial rotation angle $\theta = 45^\circ$. From top to bottom, we compare the alignment obtained with (1) MM-Align from PDB structures, (2) AlignOT using a point cloud size $n = 500$, with both point clouds and resulting density maps shown, and (3) Chimera’s fitmap function (from the density maps). (c): Alignments obtained with the pair ID 5 show the increased range of convergence of AlignOT compared with fitmap. While AlignOT successfully align the maps for $\theta = 45$ and $60^\circ$ (with the figure showing the result for $n = 1000$), it is not the case for fitmap, with a misalignment observed at $60^\circ$. (d): From top to bottom, aligning pair ID 6 with $\theta = 45^\circ$ shows a decrease in accuracy for AlignOT using a point cloud size $n = 1000$, compared with $n = 250$. We plot the corresponding the Wasserstein distance over the iterations of the algorithm, showing a larger final value for $n = 1000$, as the algorithm converges more slowly. As a result, this is one of the few cases where we found that Chimera produces a better alignment.

as for some missing parts from one map to another (pairs e and f). As illustrated in Fig. 4(a), we first used the molecular structures associated with the maps to define a ground truth alignment using the structure-based alignment function MM-align [5] (see also the Datasets Section II-E). This ground truth was used to evaluate the performance of AlignOT, as well as Chimera’s fitmap, which we used for benchmarking against our method.

Upon testing for different values of point cloud size $n \in \{250, 500, 1000\}$ and initial rotation angle $\theta \in \{45^\circ, 60^\circ, 90^\circ\}$, we recorded the angle differences between the ground truth alignment and the output using both fitmap and AlignOT, as reported in Table III. Over the 18 cases tested, AlignOT outperformed fitmap 13 times (72%), with some significant improvement observed in the majority of them (8 cases with an average improvement of more than $20^\circ$). For example, the improvement obtained in the pair ID 4 ($\theta = 45^\circ$) is visualized in Fig. 4(b), and shows how the presence of some symmetries (observed in the figure at the bottom of the molecule) can lead fitmap to misalign maps. Our experiments also confirm the improvement of the range of convergence using AlignOT, which we illustrate in Fig. 4(c) with the pair ID 5 (with good alignment obtained at $60^\circ$, contrary to fitmap). Besides, over the 5 cases where fitmap outperformed AlignOT, either the difference between the two methods is marginal (less than $2^\circ$, for ID 1, 3, 6, and $\theta = 45^\circ$), or both methods perform poorly (ID 2 with $\theta = 60, 90^\circ$). With the exception of the pair ID 2...
TABLE III
BENCHMARKING OF ALIGNOT ON PAIRS OF CONFORMATIONALLY HETEROGENEOUS MAPS, AS LISTED IN TABLE I

| ID | Angle | fitmap n = 250 | AlignOT n = 250 | AlignOT n = 500 | AlignOT n = 1000 |
|----|-------|----------------|----------------|----------------|-----------------|
| 1  | 45°   | 0.97 ± 0.71 | 2.82 ± 1.34 | 2.30 ± 1.08 | 5.73 ± 3.84 |
| 2  | 60°   | 6.39 ± 14.35 | 2.79 ± 1.32 | 2.99 ± 1.95 | 10.78 ± 8.21 |
| 3  | 90°   | 46.93 ± 25.99 | 9.22 ± 15.95 | 16.77 ± 22.09 | 32.48 ± 25.48 |
| 4  | 45°   | 37.91 ± 8.38 | 20.01 ± 34.43 | 14.99 ± 30.83 | 11.06 ± 26.68 |
| 5  | 60°   | 56.33 ± 5.79 | 62.32 ± 42.37 | 62.88 ± 41.47 | 62.58 ± 39.83 |
| 6  | 90°   | 85.21 ± 5.42 | 98.20 ± 21.54 | 99.69 ± 14.84 | 99.66 ± 13.48 |

For each pair, we ran AlignOT and fitmap and reported the angle difference between the resulting algorithm and the ground truth (see Figure 4a). For each pair, we ran our test with initial rotation angles θ ∈ {45°, 60°, 90°}, across 89 different axes covering half of the spheres S² (with 20 runs for each), and point cloud sizes n ∈ {250, 500, 1000} in AlignOT. Each table entry reports the mean and variance observed (with sample size 20), with the best results for each pair highlighted in bold. For an extended version of this table that contains the corresponding Wasserstein distances, see Supplementary Table S1.

The best results for each pair are highlighted in bold.

For poor performance from both methods, we found that the best alignment yields a Wasserstein distance that is the closest to that of the ground truth alignment, regardless of the method associated with it (for a detailed report of the distances, see Supplementary Table S1, SI File, available online). These results thus suggest that the Wasserstein distance, which determines the objective function of our method, is a more appropriate metric to use than the euclidean norm. Finally, our experiments highlight that increasing point size cloud does not necessarily lead to the best alignment from AlignOT, as a result of the trade-off between performance and speed that we also previously observed. Such a phenomenon was observed for the pair ID 6 (θ = 45°), with a better alignment for n = 250 than n = 1000, as illustrated in Fig. 4(d). As shown by the Wasserstein distance plots, the convergence is in this case significantly slower for n = 1000, which leads the algorithm to stop before the convergence condition gets achieved.

IV. DISCUSSION

In this paper, we present AlignOT, a new method for aligning cryo-EM density maps that relies on minimizing the Wasserstein distance between sampled point clouds. As shown in our experiments, AlignOT is scalable to the typical size of density maps, with a good compromise between accuracy and speed that can be achieved upon tuning the point cloud size. Our method can thus be used to quickly align maps that come from different conformations of the same protein or complex. In particular, optimizing for a transport-based metric, instead of other common metrics (e.g., overlap, correlation), allows AlignOT to generally outperform the standard local optimization method implemented in Chimera. Interestingly, the Wasserstein (or Earth-mover) distance was used in other applications in cryo-EM and tomographic projections (e.g., in interpolation or clustering [36], [37], [38], [39]), as its natural interpretation as the cost of displacing mass between two distributions makes it appropriate to compare volume-objects. In particular, the 1-Wasserstein distance was proposed after our current approach to align EM maps [40].

While the choice of the TRN algorithm to generate point clouds is justified by its previous use to represent molecular structures [10], [15], [17], it can also be replaced by any other point cloud generation method, and it would be interesting to explore how to possibly improve our method on this aspect (in particular, we could for example use Vector Quantization, as it also has been used for approximating EM maps [41], [42]). More generally, as the accuracy of alignment methods also depends on the quality of the maps, it would be interesting to combine AlignOT with recent methods that refine [43] or construct a “main chain probability map” [3] from EM maps, to generate point clouds for AlignOT. Finally, the performance of our method also depends on various hyperparameters, such as the learning rate and point cloud size values, that should be tuned to achieve the best compromise between precision and accuracy.

In the more general context of solving a rigid body alignment problem, AlignOT can be seen as a variant of the Iterative Closest Point method (ICP) framework [44], that consists of iteratively moving the point clouds according to the best way to match them. Among the many different variants of the ICP, Grave et al. similarly employed the Wasserstein distance to align language models [45]. Compared with their approach, our method differs with the use of quaternions in the gradient evaluation, as they make it possible to efficiently represent and manipulate 3D rotations. In addition, our modified version of the Sinkhorn algorithm, which evaluates the transport plan between the point clouds, provides a rapid computation of the transportation plan at each iteration, by simply reusing the vectors estimated at the previous iteration.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
As our comparative experiments indicate an improvement of the range of convergence, as well as the number of random placements leading to a successful alignment, they suggest that AlignOT can potentially provide a useful alternative from current tools (e.g., Chimera’s fitmap or VESPER [11]), that could also in the future be implemented as a plug-in for Chimera (as in our previous work [36]). Another current limitation is that the optimization problem assumes that the maps to align represent different conformations of a molecule, and thus carry approximately the same total density. However, it is also important in the context of Cryo-EM to consider the case (which VESPER does), of fitting two maps of different sizes, with one representing only a part of the other [8, 46] (making the point set incomplete for the registration problem). In this case, since we cannot simplify, as in Appendix A (SI File), available online, the rigid body alignment problem into an optimization over the rotations only, our method does not apply directly. However, as one can naturally formulate this problem in our framework as a problem of unbalanced, or partial Optimial Transport [47], and with the recent development of computational methods to solve it [48], [49], it makes such a generalization of AlignOT a promising future direction to pursue.

REFERENCES

[1] X.-C. Bai, G. McMullan, and S. H. Scheres, “How Cryo-EM is revolutionizing structural biology,” Trends Biochem. Sci., vol. 40, no. 1, pp. 49–57, 2015.
[2] F. Poitevin, A. Kushner, X. Li, and K. D. Duc, “Structural heterogeneities of the ribosome: New frontiers and opportunities for cryo-EM,” Molecules, vol. 25, no. 18, 2020, Art. no. 4262.
[3] J. He, P. Lin, J. Chen, H. Cao, and S.-Y. Huang, “Model building of protein complexes from intermediate-resolution cryo-EM maps with deep learning-guided automatic assembly,” Nature Commun., vol. 13, no. 1, 2022, Art. no. 4066.
[4] Y. Li et al., “Fine-grained alignment of cryo-electron tomograms based on MPI parallel optimization,” BMC Bioinf., vol. 20, no. 1, pp. 1–13, 2019.
[5] M. Mukherje and Y. Zhang, “MM-align: A quick algorithm for aligning multiple-chain protein complex structures using iterative dynamic programming,” Nucleic Acids Res., vol. 37, no. 11, pp. e83–e83, 2009.
[6] L. S. Moreta et al., “A probabilistic programming approach to protein structure superposition,” in Proc. IEEE Conf. Comput. Intell. Bioinf. Comput. Biol., 2019, pp. 1–5.
[7] E. F. Pettersen et al., “UCSF Chimera—A visualization system for exploratory research and analysis,” J. Comput. Chem., vol. 25, no. 13, pp. 1605–1612, 2004.
[8] P. Evans and A. McCoy, “An introduction to molecular replacement,” Acta Crystallographica Sect. D Biol. Crystallogr., vol. 64, no. 1, pp. 1–10, 2008.
[9] T. Kawabata, “Multiple subunit fitting into a low-resolution density map of a macromolecular complex using a Gaussian mixture model,” Biophysical J., vol. 95, no. 10, pp. 4643–4658, 2008.
[10] S. Birmanns and W. Wriggers, “Multi-resolution anchor-point registration of biomolecular assemblies and their components,” J. Struct. Biol., vol. 157, no. 1, pp. 271–280, 2007.
[11] X. Han, G. Terashii, C. Christoffer, S. Chen, and D. Kihara, “VESPER: Global and local cryo-EM map alignment using local density vectors,” Nature Commun., vol. 12, no. 1, pp. 1–12, 2021.
[12] G. Peyré et al., “Computational optimal transport: With applications to data science,” Found. Trends Mach. Learn., vol. 11, no. 5/6, pp. 355–607, 2019.
[13] A. Ecoffet, F. Poitevin, and K. D. Duc, “MorphOT: Transport-based interpolation between EM maps with UCSF ChimeraX,” Bioinformatics, vol. 36, no. 22,23, pp. 5528–5529, 2020.
[14] A. Ecoffet, G. Woollard, A. Kushner, F. Poitevin, and K. D. Duc, “Application of transport-based metric for continuous interpolation between cryo-EM density maps,” AIMS Math., vol. 7, no. 1, pp. 986–999, 2022.
[15] W. Wrigher, A. Milligan, K. Schult, and J. A. McCommon, “Self-organizing neural networks bridge the biomolecular resolution gap,” J. Mol. Biol., vol. 284, no. 5, pp. 1247–1254, 1998.
[16] T. Martinetz and K. Schulten, “Topology representing networks,” Neural Comput. vol. 7, no. 3, pp. 509–525, 1995.
[17] Y. Zhang et al., “State-dependent sequential allostery exhibited by chaperonin TRiC/CCT revealed by network analysis of cryo-EM maps,” Proc. Biophys. Mol. Biol., vol. 160, pp. 104–120, 2021.
[18] M. D. Zeiler, “AdaDelta: An adaptive learning rate method,” 2012, arXiv:1212.5701.
[19] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian optimization of machine learning algorithms,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2012, pp. 2951–2959.
[20] P. I. Frazier, “A tutorial on Bayesian optimization,” 2018, arXiv:1807.02811.
[21] X. Han, G. Terashi, C. Christoffer, S. Chen, and D. Kihara, “VESPER: Ribosome dynamics and tRNA movement by time-resolved electron cryomicroscopy,” Nature, vol. 466, no. 7304, pp. 329–333, 2010.
[22] X.-X. Bai, E. Rajendra, G. Yang, Y. Shi, and S. H. Scheres, “Sampling the conformational space of the catalytic subunit of human -secretase,” elife, vol. 4, 2015, Art. no. e11182.
[23] N. Fischer, A. L. Konevega, W. Wintermeyer, M. V. Rodnina, and H. Stark, “Ribosome dynamics and tRNA movement by time-resolved electron cryomicroscopy,” Nature, vol. 466, no. 7304, pp. 329–333, 2010.
[24] N. Zelesko, A. Moscovich, J. Kileel, and A. Singer, “Earthmover-based applications leading to a successful alignment, they suggest that VESPER does, of fitting two maps of different sizes, with one representing only a part of the other [8, 46] (making the point set incomplete for the registration problem). In this case, since we cannot simplify, as in Appendix A (SI File), available online, the rigid body alignment problem into an optimization over the rotations only, our method does not apply directly. However, as one can naturally formulate this problem in our framework as a problem of unbalanced, or partial Optimial Transport [47], and with the recent development of computational methods to solve it [48], [49], it makes such a generalization of AlignOT a promising future direction to pursue.
Aryan Tajmir Riahi received the BS degree in computer engineering from the Sharif University of Technology, Iran, in 2020. He is currently working toward the PhD degree in computer science with the University of British Columbia, Vancouver, Canada, where his research focuses on applications of computational optimal transport in cryo-EM.

Geoffrey Woollard received the BSc (hons) degree in biophysics from the University of British Columbia in Vancouver, Canada, in 2011, and the MSc degree in genome science and technology from the University of British Columbia, in 2014. He is currently working toward the PhD degree in computer science with the University of British Columbia. His research focuses on forward models and inference for single particle cryogenic electron microscopy (cryo-EM). From 2016–17, he was an associate scientist with Cyclicia Inc. (Toronto), where he developed applied machine learning approaches on structural data of small molecule-protein complexes for drug discovery and development applications. In 2018, he was an associate scientist with Structura Biotechnology Inc. (Toronto), where he worked on cryoSPARC, a scientific software platform for (cryo-EM) used in research and drug discovery pipelines.

Frédéric Poitevin received the BS and MS degrees in chemistry from the Université Paris 6, Paris, France, in 2000/2009, the diploma in chemistry from Ecole Normale Supérieure, Paris, in 2010, and the PhD degree in bioinformatics, genome analysis and modeling from the Université Paris Diderot, Paris, France, in 2013, for his work carried with the Pasteur Institute, Paris, in the “Structural Dynamics of Macromolecules” group led by Marc Delarue, his PhD advisor. From 2013 to 2015, he was a postdoctoral scholar with the Laboratory for Theoretical Physics, IPhtCEA, Saclay, France, with Marc Delarue and Henri Orland. From 2015 to 2020, he was a postdoctoral scholar with Stanford University in the Computational Structural Biology Group led by Michael Levitt. He is currently a staff associate scientist with SLAC National Accelerator Laboratory, in the LCLS Department, where he streamlines data processing workflows while developing computational methods in Structural Biology, mainly aimed at extracting molecular dynamics information from various types of experimental data (X-ray Crystallography, Cryogenic Electron Microscopy, etc.).

Anne Condon received the BS degree in mathematics and computer science from University College Cork, Ireland, in 1982, and the PhD degree in computer science from the University of Washington, in 1987. She was a faculty member at the University of Wisconsin from 1987 to 1999. Since then, she has been a professor with the University of British Columbia, where she has served as department head and associate dean in the Faculty of Science. Her research in computational complexity theory and algorithms currently focuses on DNA programming—the art and science of writing programs that are realized and executed by DNA. She has received an ACM Distinguished Dissertation Award, an NSF Young Investigator Award, the Anita Borg Institute Technical Leadership Award, and Distinguished Alumni Awards from University College Cork and U. Washington’s CSE Department and College of Engineering. She is a fellow of the Royal Society of Canada and the Association for Computing Machinery.

Khanh Dao Duc received the BS degree in mathematics from the Ecole Normale Supérieure de Lyon, Lyon, France, in 2006, the MS degree in mathematics from the Université Paris 6, Paris, France, in 2009, and the PhD degree in applied mathematics from the Ecole Normale Supérieure, Paris, in 2013. From 2014 to 2015, and 2017 to 2019, he was a postdoctoral fellow with the University of California, Berkeley, CA, USA. From 2015 to 2017, he was a Simons postdoctoral fellow with the University of Pennsylvania, Philadelphia, USA. He is currently an assistant professor of mathematics with the University of British Columbia, Vancouver, BC, Canada, where his research group focuses on developing mathematical and computational methods for various problems in structural, molecular and cell biology, and neuroscience.