
3D alignment of cryogenic electron microscopy density maps by minimizing their Wasserstein distance

Aryan Tajmir Riahi

Department of Computer Science
University of British Columbia
Vancouver, BC V6T 1Z4
artajmir@cs.ubc.ca

Geoffrey Woollard

Department of Computer Science
University of British Columbia
Vancouver, BC V6T 1Z4
gw@cs.ubc.ca

Frédéric Poitevin

Department of LCLS Data Analytics
SLAC National Accelerator Laboratory
Menlo Park, CA 94025
frederic.poitevin@stanford.edu

Anne Condon

Department of Computer Science
University of British Columbia
Vancouver, BC V6T 1Z4
condon@cs.ubc.ca

Khanh Dao Duc

Department of Mathematics
University of British Columbia
Vancouver, BC V6T 1Z4
kdd@math.ubc.ca

Abstract

Aligning electron density maps of multiple conformations of a biomolecule from Cryogenic electron microscopy (cryo-EM) is a first key step to study conformational heterogeneity. As this step remains challenging, with standard alignment tools being potentially stuck in local minima, we propose here a new procedure, 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 show that our method outperforms standard methods on experimental data, with an increased range of rotation angles leading to proper alignment, suggesting that it can be further applied to align 3D EM maps.

1 Introduction

1.1 Background

Solving the 3D structures of biomolecules is key to understanding 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 EM density maps, i.e., large 3D grids of voxels with different levels of intensities (typically $\sim 100^3$ to $\sim 500^3$ voxels), aligning these maps is needed to further compare them.

Efficient methods have been developed to align two protein structures [3, 4], 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.

1.2 Main contributions

In this paper, we introduce a novel approach, called **AlignOT**, for 3D alignment of density maps. More precisely, using a point-cloud representation of 3D maps, AlignOT uses a stochastic gradient algorithm that minimizes the so-called Wasserstein distance between two maps. This non-Euclidean distance is associated with the theory of Optimal Transport (OT) [5], with recent advances that make tractable the computation of transport-based distances [6, 7]. After describing the procedure in detail, we run it on experimental data to quantify its accuracy, and show that it outperforms standard local alignment methods.

1.3 Related work

To solve the rigid body alignment problem for 3D cryo-EM density maps, standard approaches use various algorithms to maximize correlation [8, 9, 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 the more general context of solving a rigid body alignment problem, the Iterative Closest Point method (ICP) [12], that consists of iteratively moving the point clouds according to the best way to match them, provides a framework for more recent methods [13]. Among these variants of the ICP, Grave *et al.* employed the Wasserstein distance to align language models [14], but has not been applied to the context of 3D rotations.

2 Methods

2.1 Background on Optimal Transport and Wasserstein distance

We use a non-Euclidean metric that derives from the theory of Optimal Transport [5]. For two given point clouds, $\mathbf{A} = \{a_1, \dots, a_n\}$ and $\mathbf{B} = \{b_1, \dots, b_n\}$, 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 \mathbf{A} and \mathbf{B} , denoted by $\mathcal{W}_{2,\epsilon}(\mathbf{A}, \mathbf{B})$, is then defined as

$$\mathcal{W}_{2,\epsilon}(\mathbf{A}, \mathbf{B}) = \left[\min_{P \in \mathbb{R}_+^{n \times n}} \sum_{i,j=1}^n C_{i,j} P_{i,j} + \epsilon H(P) \right]^{1/2}, \quad (1)$$

s.t. $P \cdot \mathbb{1} = P^T \cdot \mathbb{1} = \mathbb{1}/n$

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

$$H(P) = \sum_{i,j=1}^n P_{i,j} \log P_{i,j}. \quad (2)$$

The minimizer of equation (1) is called the *transport plan*. For the rest of the Methods section, we will simply denote the Wasserstein distance as $\mathcal{W}_{2,\epsilon}$, and $P_{i(a),i(b)}$ as $P_{a,b}$, where $i(a)$ and $i(b)$ are the indices of the two points a and b in \mathbf{A} and \mathbf{B} , respectively.

2.2 3D map alignment with computational Optimal Transport

To align two 3D EM maps (\mathcal{A} and \mathcal{B}), we first represent them by point clouds i.e., a set of n points $\in \mathbb{R}^3$ (\mathbf{A} and \mathbf{B}). To do so, we use the topology representing network algorithm (TRN) [15] which

has been previously applied to EM density maps [16]. From the point cloud representations \mathbf{A} and \mathbf{B} , we solve the optimization problem

$$q_{\text{opt}} = \underset{q \in \mathbb{H}}{\operatorname{argmin}} \mathcal{W}_{2,\epsilon}(R_q(\mathbf{A}), \mathbf{B}), \quad (3)$$

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 $R_q(\mathbf{A}) = \{R_q(a_i) | a_i \in \mathbf{A}\}$. We explain in Appendix A 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. Our stochastic gradient descent procedure to solve (3), called AlignOT, is detailed in Algorithm 1 in Appendix C. At each iteration, the algorithm updates q from the transport plan P between $R_q(\mathbf{A})$ and \mathbf{B} as follows: After sampling one point $a \in R_q(\mathbf{A})$, we evaluate $\pi(a) = \operatorname{argmax}_{b \in \mathbf{B}} P_{a,b}$, and compute the gradient in q associated with $d(\pi(a), a)^2$, where d is the Euclidean distance, to update q . To compute the transport plan, we apply the Sinkhorn algorithm [6], with the initial vectors set as the outputs of the previous iteration. 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. With the most limiting part of this algorithm being the evaluation of the OT plan matrix, the overall time complexity of the algorithm is $O(n^2 L \log n \epsilon^{-3})$, where n is the size of the point cloud.

The hyperparameters of this procedure are the learning rate α associated with gradient descent, the regularization parameter ϵ associated with the Wasserstein distance, and a threshold δ 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.

2.3 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 [17]. 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 will be available upon publication.

2.4 Dataset

To evaluate our method, we tested AlignOT on aligning two point clouds obtained from a conformationally heterogeneous pair of density maps. To do so, we used two atomic cryo-EM structures of Hsp90-Cdc37-Cdk4 complex (PDB:5fwl and PDB:5fwm) [18], shown in Figure 1. We applied *MM-align* [3] to define a ground truth alignment. 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. Finally we used Chimera’s `molmap` command to generate density maps from aligned structures.

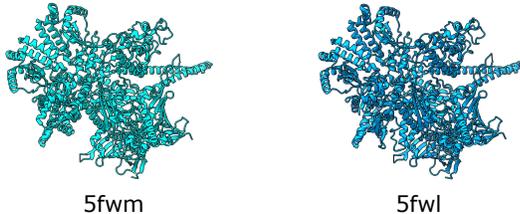


Figure 1: 3D maps used in our experiments, visualized with Chimera [8], representing different conformational states of Hsp90-Cdc37-Cdk4 complex.

3 Results

3.1 Alignment of maps from different conformations

To test AlignOT on our dataset (see Section 2.4), we sampled two clouds of 500 points, and applied a rotation defined in its axis-angle representation by an arbitrary axis, and an angle $\theta = 50^\circ$. Figure 2a 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 Figure 2b, with all the four trajectories converging

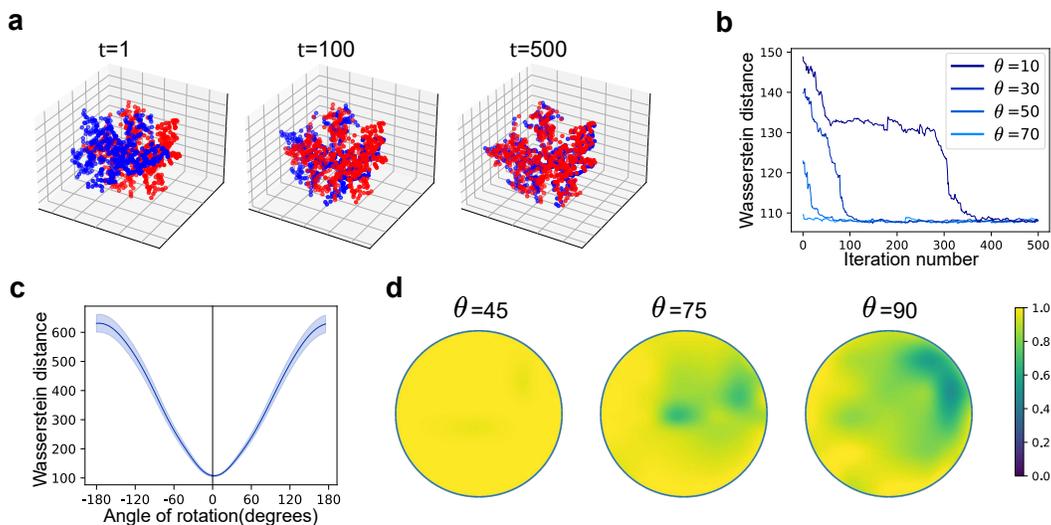


Figure 2: Alignment of two point clouds obtained from two conformationally heterogeneous complexes (PDB:5fwm and PDB:5fwl) using AlignOT **(a)**: We generated two point clouds of aligned structures that differ from a 3D rotation, and applied AlignOT. The figure shows the result of the procedure at different iterations ($t = 1, 100, 500$). The blue and red dots represent the rotated and target point clouds, respectively. **(b)**: For different initial values $\theta (= 10^\circ, 30^\circ, 50^\circ, 70^\circ)$ of the angle difference between the two maps and same rotation axis, we plot the Wasserstein distance between the two target and rotated point clouds across the iterations of the algorithm, showing that they all converge to the same limit. **(c)**: For the same rotation axis as in **(b)**, we plot the average Wasserstein distance between two sampled point clouds as a function of θ , for point clouds of size 500. Error regions show the standard deviation from sampling 100 different point clouds for each angle. **(d)**: Alignment success rate of AlignOT at fixed angles $\theta = 45^\circ, 75^\circ$, and 90° , and over the rotation axes that cover the upper hemishpere of S^2 . Heatmaps show the percentage of outcomes that result in an alignment with error $\leq 5^\circ$, where each point of the disk is the projection of the axis considered in S^2 (89 in total, with 20 runs for each).

to the same value and resulting in a successful alignment. However, we also observed that as θ 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 θ . To interpret these results, we further plotted in Figure 2c how the Wasserstein distance varies on average (after sampling different point clouds of size 500), as a function of θ (and same rotation axis). This observation shows a strong global minimum at $\theta = 0$ with no significant local minima which foreshadows AlignOT good results, however, the low magnitude of gradient for rotations with high angle can be problematic for the convergence. We finally evaluated the probability to successfully align the maps for initial rotations of fixed angle θ ($45^\circ, 75^\circ$ and 90°), and across different axes covering half of the sphere S^2 . Upon mapping the axes on the planar disk in Figure 2d, we found local regions of poorer alignment. 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 75° (see Figure 2d), with the performance worsening as θ increases. On average, each of these alignments by AlignOT took 16.36 seconds on an Intel(R) Core(TM) workstation with i7-7700HQ CPU @ 2.80GHz 2.81 GHz with 16.0 GB RAM.

3.2 Benchmarking against Chimera `fitmap` alignment function

As mentioned in the previous section there is a limiting range within which the method can align two maps. We further investigate this range for AlignOT and compare it with Chimera’s `fitmap` local search, which performs a steepest ascent optimization to align maps according to their overlapping score [8, 19]. To do so, we used initial rotations with a fixed axis and across different angle $0^\circ \leq \theta \leq 180^\circ$, and then applied AlignOT and `fitmap`. As shown in Figure 3, AlignOT outperforms `fitmap` local search with an increased range of convergence (from $\sim 55^\circ$ to $\sim 115^\circ$ in this experiment).

Under more general conditions, we also performed a comparison with `fitmap` using different values of point cloud size $n \in \{250, 500, 1000\}$ and initial rotation angle $\theta \in \{45^\circ, 60^\circ, 90^\circ\}$, with 89 rotation axes that cover the upper hemisphere of S^2 . The angle differences between the ground truth alignment and the output, as reported in Table 1, show that over all three test cases, AlignOT outperformed `fitmap` on average, with some significant improvement observed as θ increases. Our experiments also confirm the improvement of the range of convergence using AlignOT, which we illustrated in Figure 3. These results suggest that the Wasserstein distance, which determines the objective function of our method, is a more appropriate metric to use than the Euclidean norm.

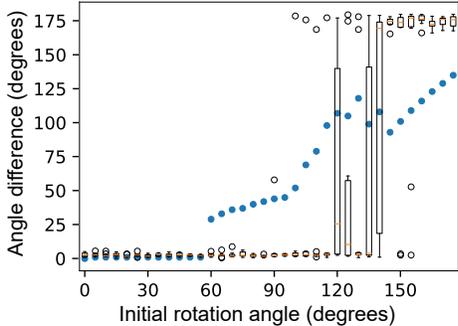


Figure 3: Comparison between AlignOT and `fitmap`: With a fixed axis and for a rotation angle difference $\theta \in (0, 180)$ degrees, we run AlignOT and `fitmap` to align maps from PDB:5fwm and PDB:5fwl. The blue dots show the error obtain in the alignment using `fitmap`, while the box plot (minimum, first quartile, median, third quartile, and maximum over 10 runs) the error of AlignOT.

Table 1: Benchmarking of AlignOT on maps listed in Section 2.4. Using initial rotation angles $\theta \in \{45^\circ, 60^\circ, 90^\circ\}$, across 89 different axes covering half of the sphere S^2 (with 20 runs for each), and point cloud sizes $n \in \{250, 500, 1000\}$ we ran AlignOT and `fitmap` and recorded the angle difference between the resulting algorithm and the ground truth. Finally, we reported the mean and the standard deviation of recorded angle differences among all experiments with the same method and the same initial angle, with the best results for each angle highlighted in bold.

Angle	fitmap	AlignOT		
		$n = 250$	$n = 500$	$n = 1000$
45	2.02	5.94	3.77	2.02
	± 5.41	± 15.79	± 11.68	\pm 1.22
60	26.10	9.37	6.64	3.37
	± 19.97	± 28.47	± 24.80	\pm 14.33
90	60.52	17.454	26.27	16.71
	± 16.78	± 45.22	± 58.88	\pm 47.56

4 Discussion

In this paper, we present 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, and can 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 [20, 21, 22, 23]), as its natural interpretation as the cost of displacing mass between two distributions makes it appropriate to compare volume-objects.

From the current results, it would be interesting to explore how to possibly improve our method on several aspects. While the choice of the TRN algorithm to generate point clouds is justified by its previous use to represent molecular structures [24], 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 [25, 26]). A more thorough study would also be needed to study the impact various hyperparameters of AlignOT (e.g. learning rate and point cloud size values), and how to fine tune them to achieve the best compromise between speed and accuracy. Finally, one can

extend the present alignment problem to the important case of fitting two maps of different sizes, with one representing only a part of the other [9, 27]. This problem can be naturally formulated in our framework as a problem of *unbalanced*, or *partial* Optimal Transport [28]. The recent development of computational methods to solve it [29, 30] makes such a generalization of AlignOT another promising future direction to pursue.

References

- [1] Xiao-Chen Bai, Greg McMullan, and Sjors HW Scheres. How Cryo-EM is revolutionizing structural biology. *Trends in Biochemical Sciences*, 40(1):49–57, 2015.
- [2] Frédéric Poitevin, Artem Kushner, Xinpei Li, and Khanh Dao Duc. Structural heterogeneities of the ribosome: new frontiers and opportunities for cryo-em. *Molecules*, 25(18):4262, 2020.
- [3] Srayanta Mukherjee and Yang Zhang. MM-align: a quick algorithm for aligning multiple-chain protein complex structures using iterative dynamic programming. *Nucleic acids research*, 37(11):e83–e83, 2009.
- [4] Lys Sanz Moreta, Ahmad Salim Al-Sibahi, Douglas Theobald, William Bullock, Basile Nicolas Rommes, Andreas Manoukian, and Thomas Hamelryck. A probabilistic programming approach to protein structure superposition. In *2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–5. IEEE, 2019.
- [5] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.
- [6] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.
- [7] Jason Altschuler, Jonathan Weed, and Philippe Rigollet. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. *arXiv preprint arXiv:1705.09634*, 2017.
- [8] Eric F Pettersen, Thomas D Goddard, Conrad C Huang, Gregory S Couch, Daniel M Greenblatt, Elaine C Meng, and Thomas E Ferrin. UCSF chimera—a visualization system for exploratory research and analysis. *Journal of Computational Chemistry*, 25(13):1605–1612, 2004.
- [9] Philip Evans and Airlie McCoy. An introduction to molecular replacement. *Acta Crystallographica Section D: Biological Crystallography*, 64(1):1–10, 2008.
- [10] Takeshi Kawabata. Multiple subunit fitting into a low-resolution density map of a macromolecular complex using a gaussian mixture model. *Biophysical journal*, 95(10):4643–4658, 2008.
- [11] Xusi Han, Genki Terashi, Charles Christoffer, Siyang Chen, and Daisuke Kihara. VESPER: global and local cryo-EM map alignment using local density vectors. *Nature communications*, 12(1):1–12, 2021.
- [12] Paul J Besl and Neil D McKay. Method for registration of 3-D shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992.
- [13] François Pomerleau, Francis Colas, Roland Siegwart, and Stéphane Magnenat. Comparing icp variants on real-world data sets. *Autonomous Robots*, 34(3):133–148, 2013.
- [14] Edouard Grave, Armand Joulin, and Quentin Berthet. Unsupervised alignment of embeddings with Wasserstein procrustes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1880–1890. PMLR, 2019.
- [15] Thomas Martinetz and Klaus Schulten. Topology representing networks. *Neural Networks*, 7(3):507–522, 1994.

- [16] Yan Zhang, James Krieger, Karolina Mikulska-Ruminska, Burak Kaynak, Carlos Oscar S. Sorzano, José María Carazo, Jianhua Xing, and Ivet Bahar. State-dependent sequential allostery exhibited by chaperonin TRiC/CCT revealed by network analysis of Cryo-EM maps. *Progress in Biophysics and Molecular Biology*, 160:104–120, 2021.
- [17] She Zhang, James M Krieger, Yan Zhang, Cihan Kaya, Burak Kaynak, Karolina Mikulska-Ruminska, Pemra Doruker, Hongchun Li, and Ivet Bahar. ProDy 2.0: increased scale and scope after 10 years of protein dynamics modelling with Python. *Bioinformatics*, 37(20):3657–3659, oct 2021.
- [18] Kliment A Verba, Ray Yu-Ruei Wang, Akihiko Arakawa, Yanxin Liu, Mikako Shirouzu, Shigeyuki Yokoyama, and David A Agard. Atomic structure of hsp90-cdc37-cdk4 reveals that hsp90 traps and stabilizes an unfolded kinase. *Science*, 352(6293):1542–1547, 2016.
- [19] Thomas D Goddard, Conrad C Huang, Elaine C Meng, Eric F Pettersen, Gregory S Couch, John H Morris, and Thomas E Ferrin. Ucsf chimeraX: Meeting modern challenges in visualization and analysis. *Protein Science*, 27(1):14–25, 2018.
- [20] Arthur Ecoffet, Frédéric Poitevin, and Khanh Dao Duc. Morphot: Transport-based interpolation between em maps with ucsf chimeraX. *Bioinformatics*, 36(22-23):5528–5529, 2020.
- [21] Arthur Ecoffet, Geoffrey Woollard, Artem Kushner, Frédéric Poitevin, and Khanh Dao Duc. Application of transport-based metric for continuous interpolation between cryo-em density maps. *AIMS Mathematics*, 7(1):986–999, 2022.
- [22] Nathan Zelesko, Amit Moscovich, Joe Kileel, and Amit Singer. Earthmover-based manifold learning for analyzing molecular conformation spaces. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pages 1715–1719. IEEE, 2020.
- [23] Rohan Rao, Amit Moscovich, and Amit Singer. Wasserstein k-means for clustering tomographic projections. *arXiv preprint arXiv:2010.09989*, 2020.
- [24] Yan Zhang, James Krieger, Karolina Mikulska-Ruminska, Burak Kaynak, Carlos Oscar S Sorzano, José-María Carazo, Jianhua Xing, and Ivet Bahar. State-dependent sequential allostery exhibited by chaperonin TRiC/CCT revealed by network analysis of Cryo-em maps. *Progress in Biophysics and Molecular Biology*, 160:104–120, 2021.
- [25] Pedro A De-Alarcón, Alberto Pascual-Montano, Amarnath Gupta, and Jose M Carazo. Modeling shape and topology of low-resolution density maps of biological macromolecules. *Biophysical Journal*, 83(2):619–632, 2002.
- [26] Slavica Jonić and Carlos Oscar S Sorzano. Versatility of approximating single-particle electron microscopy density maps using pseudoatoms and approximation-accuracy control. *BioMed Research International*, 2016, 2016.
- [27] Takeshi Kawabata. Rigid-body fitting of atomic models on 3d density maps of electron microscopy. *Integrative Structural Biology with Hybrid Methods*, pages 219–235, 2018.
- [28] Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. Unbalanced optimal transport: geometry and kantorovich formulation. 2015.
- [29] Patrice Koehl, Marc Delarue, and Henri Orland. Physics approach to the variable-mass optimal-transport problem. *Physical Review E*, 103(1):012113, 2021.
- [30] Khang Le, Huy Nguyen, Khai Nguyen, Tung Pham, and Nhat Ho. On multimarginal partial optimal transport: Equivalent forms and computational complexity. In *International Conference on Artificial Intelligence and Statistics*, pages 4397–4413. PMLR, 2022.

Appendixes

A Optimal translation for the rigid body alignment problem

To formalize the rigid body alignment problem, assuming that the EM maps are represented by two 3D point clouds $\mathbf{A} = \{a_1, \dots, a_n\}$ and $\mathbf{B} = \{b_1, \dots, b_n\}$, 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

$$\mathcal{L}_d(R, T) = d(\text{move}_{R,T}(\mathbf{A}), \mathbf{B})^2, \quad (4)$$

where $\mathcal{L}_d(R, T)$ is defined over rotation matrices $R \in SO(3)$ and translation vectors $T \in \mathbb{R}^3$, and the operator $\text{move}_{R,T}(\mathbf{A})$ is defined as

$$\text{move}_{R,T}(\mathbf{A}) = \{Ra_i + T | a_i \in \mathbf{A}\}. \quad (5)$$

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 [5]. 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 equation (1) in the Methods section 2.1), denoted $\mathcal{W}_{2,\epsilon}$. Besides, given the centers of mass $\bar{a} = \frac{1}{n} \sum_{i=1}^n a_i$ and $\bar{b} = \frac{1}{n} \sum_{i=1}^n b_i$, and the centered point clouds $\mathbf{A}_c = \{a_{c_i} = a_i - \bar{a} | a_i \in \mathbf{A}\}$ and $\mathbf{B}_c = \{b_{c_i} = b_i - \bar{b} | b_i \in \mathbf{B}\}$, we can show that the optimal translation of the objective function (4) is

$$T_{\text{opt}} = \bar{b} - R_{\text{opt}}\bar{a}, \quad (6)$$

where

$$R_{\text{opt}} = \underset{R \in SO(3)}{\text{argmin}} \mathcal{W}_{2,\epsilon}(R(\mathbf{A}_c), \mathbf{B}_c). \quad (7)$$

To do so, using previous definitions equation (4) yields

$$\mathcal{L}(R, T) = \mathcal{W}_{2,\epsilon}(\text{move}_{R,T}(\mathbf{A}), \mathbf{B})^2 \quad (8)$$

$$= \min_{P \in \mathbb{R}_+^{n \times n}} \sum_{i,j=1}^n \|Ra_i + T - b_j\|_2^2 P_{i,j} + \epsilon H(P) \quad (9)$$

(s.t. $\forall 1 \leq j \leq n : \sum_{i=1}^n P_{i,j} = \sum_{i=1}^n P_{j,i} = \frac{1}{n}$)

$$= \min_{P \in \mathbb{R}_+^{n \times n}} \sum_{i,j=1}^n \|Ra_{c_i} + R\bar{a} + T - b_{c_j} - \bar{b}\|_2^2 P_{i,j} + \epsilon H(P), \quad (10)$$

where we used definition of the entropy regularized Wasserstein distance from Equation (1). This further simplifies as

$$\begin{aligned} \mathcal{L}(R, T) &= \min_{P \in \mathbb{R}_+^{n \times n}} \sum_{i,j=1}^n \|Ra_{c_i} - b_{c_j}\|_2^2 P_{i,j} + \sum_{i,j=1}^n \|R\bar{a} + T - \bar{b}\|_2^2 P_{i,j} \\ &+ \sum_{i,j=1}^n (Ra_{c_i} - b_{c_j}) \cdot (R\bar{a} + T - \bar{b}) P_{i,j} + \epsilon H(P) \end{aligned} \quad (11)$$

$$= \min_{P \in \mathbb{R}_+^{n \times n}} \sum_{i,j=1}^n \|Ra_{c_i} - b_{c_j}\|_2^2 P_{i,j} + \sum_{i,j=1}^n \|R\bar{a} + T - \bar{b}\|_2^2 P_{i,j} + \epsilon H(P) \quad (12)$$

$$= \min_{P \in \mathbb{R}_+^{n \times n}} \left[\sum_{i,j=1}^n \|Ra_{c_i} - b_{c_j}\|_2^2 P_{i,j} + \epsilon H(P) \right] + \|R\bar{a} + T - \bar{b}\|_2^2 \quad (13)$$

where in (11) we used the fact that $\sum_{i,j=1}^n P_{i,j} a_{c_i} = \frac{1}{n} \sum_{i=1}^n a_{c_i} = \bar{0}$ and $\sum_{i,j=1}^n P_{i,j} b_{c_j} = \frac{1}{n} \sum_{j=1}^n b_{c_j} = \bar{0}$. Also, we used the fact that $\sum_{i,j=1}^n P_{i,j} = 1$ in (12). The second term in Equation (13) is minimized for $T = \bar{b} - R\bar{a}$, i.e. the translation that aligns the two centers of mass. Thus, the search for an optimal rigid body transformation in (7) can be simplified to rotations after matching the centers of mass of \mathbf{A}, \mathbf{B} , leading to equation (3) of the Methods section 2.1.

B Quaternion Representation of 3D rotations

To formalize 3D rotations in *AlignOT* we use the quaternion representation (\mathbb{H}). In this representation, given a point $a = (x, y, z) \in \mathbb{R}^3$ and a rotation with angle θ around axis $\vec{u} = (u_x, u_y, u_z)$, we form quaternions

$$q = \cos \theta/2 + u_x \sin \theta/2i + u_y \sin \theta/2j + u_z \sin \theta/2k, \quad (14)$$

$$p = xi + yj + zk, \quad (15)$$

where i, j, k are the basic quaternions such that

$$i^2 = j^2 = k^2 = ijk = -1. \quad (16)$$

Using equation (16) we compute the following term

$$x_{\text{res}}i + y_{\text{res}}j + z_{\text{res}}k = qpq^*, \quad (17)$$

where $R_q(a) = (x_{\text{res}}, y_{\text{res}}, z_{\text{res}}) \in \mathbb{R}^3$ is the coordinates of a after rotation, and q^* is the conjugate of q and is defined as

$$(q_0 + q_1i + q_2j + q_3k)^* = q_0 - q_1i - q_2j - q_3k. \quad (18)$$

We then define the absolute norm of q as

$$\|q\|^2 = qq^* = q_0^2 + q_1^2 + q_2^2 + q_3^2 \in \mathbb{R}. \quad (19)$$

In *AlignOT*, we also compute the gradient associated with the function $f(q) = f(q_0, q_1, q_2, q_3) = \|R_q(a) - b\|_2^2$, where $a, b \in \mathbb{R}^3$ as

$$\nabla f = \frac{\partial f}{\partial q_0} + \frac{\partial f}{\partial q_1}i + \frac{\partial f}{\partial q_2}j + \frac{\partial f}{\partial q_3}k.$$

C Alignment procedure

Algorithm 1 AlignOT: 3D density maps alignment with SGD using unit quaternions and Wasserstein distance

Input two 3D density maps \mathcal{A}, \mathcal{B} , number of sampled points $n \in \mathbb{R}$, learning rate $\alpha \in \mathbb{R}$, regularization parameter $\epsilon \in \mathbb{R}$, maximum number of iterations $L \in \mathbb{N}$, and gradient threshold $\delta \in \mathbb{R}$

- 1: Sample two sets of n points $\mathbf{A}, \mathbf{B} \subset \mathbb{R}^3$ from \mathcal{A}, \mathcal{B} respectively, using TRNs
 - 2: $q = 1 + 0i + 0j + 0k$
 - 3: $G = \alpha^2$
 - 4: **while** not converged **and** the number of iterations is at most L **do**
 - 5: Compute $R_q(\mathbf{A})$
 - 6: Compute P to be the OT plan matrix between $R_q(\mathbf{A})$ and \mathbf{B}
 - 7: Randomly select $a \in R_q(\mathbf{A})$
 - 8: $b = \pi(a)$
 - 9: $f(q) = d(R_q(a), b)^2$ (where d is Euclidean distance in \mathbb{R}^3)
 - 10: $G = G + \|\nabla f(q)\|^2$
 - 11: $q = q - \frac{\alpha}{\sqrt{G}} \times \nabla f(q)$
 - 12: $q = \frac{q}{\|q\|}$
 - 13: **end while**
 - 14: **return** q
-