Comparing Morse Complexes Using Optimal Transport:
An Experimental Study
Supplementary Materials

Mingzhe Li[*]   Carson Storm[†]   Austin Yang Li[‡]   Tom Needham [§]   Bei Wang [¶]
University of Utah   University of Utah   University of Utah   Florida State University   University of Utah

A DISCUSSION ON OT-TYPE DISTANCES

We use an example in Fig. 1 to explain the intuition behind the Wasserstein distance, GW distance, and FGW distance. We consider two scalar fields shown on the left, with corresponding Morse graphs shown on the right. To compute OT-type distances, we use the geometric locations of nodes as attributes, and the shortest geodesics between nodes to describe the network structures; see Sec. 4 for details.
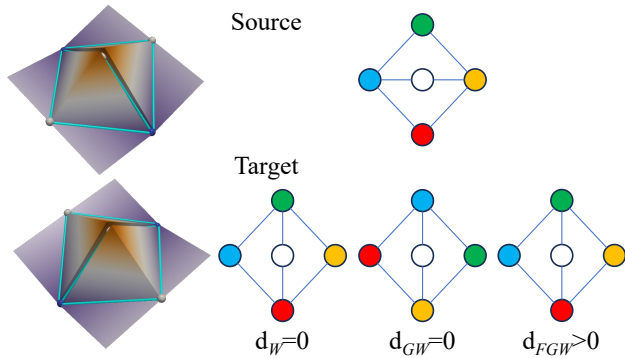

Figure 1: Left: synthetic scalar fields, together with their corresponding Morse graphs (in cyan). Right: Morse graphs of the scalar fields; colormap shows structural correspondences under various OT-type metrics.

We observe the behavior of three OT-type distances based on the node color correspondences. The Wasserstein distance is 0 because the locations of matched nodes in the source and target are identical. The GW distance is also 0 since it finds a coupling such that the graph structure in the target is identical to the source. Such a coupling rotates the node color correspondence by $90°$. However, since the FGW distance takes both graph structures and geometric locations into consideration, it becomes non-zero in this example.

B DETAILS ON DATASETS

The synthetic Sinusoidal dataset is formed by the sum of two sine waves. The target field is generated with additional noise in comparison with the source field.

The Wind dataset includes 15 wind velocity fields from the IRI/LDEO Climate Data Library: *pressure_level_wind* is obtained via the NCEP CFSv2 Ensemble [18] with forecasted and perturbed parameters. It focuses on a spatial range of $150°W$-$49.5°W$ and

---

[*]e-mail: mingzhe.li@utah.edu

[†]e-mail: u1120712@utah.edu

[‡]e-mail: u1364758@utah.edu

[§]e-mail: tneedham@fsu.edu

[¶]e-mail: beiwang@sci.utah.edu

$90°N$-$10°S$ at a pressure level 200 hPA and a forecast hour 0 on January 01, 2015.

The HeatedCylinder dataset [1] comes from a simulation via the Gerris flow solver [15]. The simulation shows a 2D flow of a heated cylinder using the Boussinesq approximation [9]. It describes a time-varying turbulent plume containing multiple small vortices. We select 100 time steps from the original dataset (800-899 from 2000 time steps) and compute the Morse graphs from the velocity magnitude fields.

The NavierStokes dataset is a direct numerical Navier Stokes simulation available from [12]. Camarri et al. [2] used a version of the simulation that Tino Weinkauf had uniformly resampled, which was used for smoke visualization by von Funck et al. [19]. The dataset is a 3D time-varying fluid flow simulation around a square cylinder placed symmetrically between two parallel walls. We select time steps 60-66 and a 2D slice perpendicular to the z-axis ($z = 24$) and compute Morse graphs from the velocity magnitude fields.

The RedSea dataset comes from the IEEE 2020 SciVis Contest [10] and shows an eddy simulation of the Red Sea. It uses the MIT ocean general circulation model (MITgcm) and the Data Research Testbed (DART) [11] to create an ensemble with varying initial conditions. A 3D area with a resolution of $500 \times 500 \times 50$ is sampled across 60 time steps [16]. We use 10 ensemble members from 2D slices perpendicular to the z-axis ($z = 1$) at time step 40. We compute the Morse graphs from the velocity magnitude fields.

C PREPROCESSING AND PARAMETER SETTING

Persistence simplification. We apply *persistence simplification* to each dataset before computing the Morse graphs. We normalize the range of a given scalar field to be $[0, 1]$ and use $\varepsilon \in [0, 1]$ to denote the simplification threshold. ε is chosen based on the *persistence graph* [8], where the x-axis represents ε, the y-axis captures the number of local maxima (in our setting), and a plateau implies a stable range of scales to separate features from noise. An ε-simplification means that the scalar field is simplified such that critical points with persistence less than ε are removed.

For the Sinusoidal dataset, Fig. 2 (left) shows the simplification threshold $\varepsilon = 7\%$ chosen at a plateau for both the source field (blue) and the target field (orange). The parameter ε is chosen to be $3\%, 10\%, 7\%$, and 1% for the Wind, HeatedCylinder, NavierStokes, and RedSea datasets, respectively.

Parameter setting for FGW distance. In this paper, the attribute space is set to be $\mathbb{R}^2$ endowed with a Euclidean distance, and the attribute function assigns a node its location in the domain. We apply normalization such that the Euclidean distance between a node in the source Morse graph and a node in the target Morse graph falls within $[0, 1]$. We also rescale elements in each network function matrix W such that they fall into the range $[0, 1]$. With such a normalization, we set $\alpha = 0.5$ when computing the FGW and pFGW distances, striking a balance between the Wasserstein component and the GW component.

Node probability distribution. We assign a probability distribution to all nodes in the Morse complexes when we model them as measure
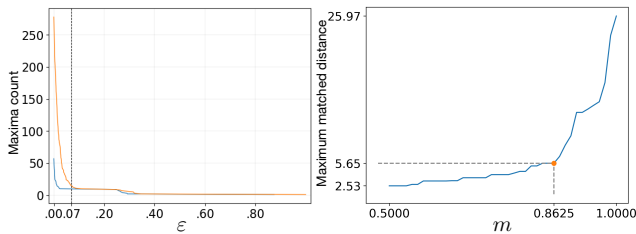
Figure 2: Parameter tuning for the Sinusoidal dataset. Left: persistence simplification using a persistence graph, where $\varepsilon = 7\%$. Right: $m = 0.8625$ is the elbow point based on the maximum matched distances.

networks. Without any prior knowledge about the space, a uniform probability distribution is a reasonable option for computing similarities between spaces based on previous works (e.g. [5,7,13,14]), since it ensures that all nodes in the space have equal importance in the distance.
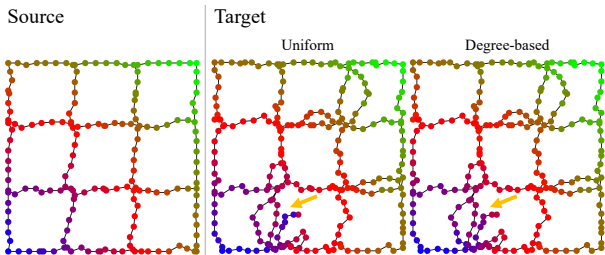

Figure 3: FGW distance for the Sinusoidal dataset with different node probability distributions.

In this paper, we apply a probability distribution that is proportional to the degree of nodes instead of the uniform distribution. If we apply a uniform distribution, due to our node sampling strategy, the number of nodes along each edge may be sensitive to minor changes in the geometry of an edge, thus affecting the distribution. Furthermore, newly appeared or disappeared edges along with their sampled nodes may also affect the probability of existing nodes. By applying a degree-based probability distribution, nodes with higher degrees (typically critical points) serve as anchor points that are considered more important than the (sampled) regular points; this approach has also been prevalent in the graph OT literature, due to its superior empirical performance for certain tasks (e.g., [6,20]). For example, in Fig. 3, the result using the FGW distance with uniform distribution shows undesired matching as indicated by the orange arrow. The degree-based distribution improves the coupling in the same area, showing a smoother color trend.

Partial OT parameter tuning. We explain the parameter tuning of partial OT via the Sinusoidal example in Fig. 4, following a previous work [13]. As the parameter m decreases from 1.00 to 0.80, the number of hollow nodes in the target graph (e.g., nodes that are ignored during the coupling) increases gradually. During this process, nodes in the target graph on the noisy edges are ignored first, as matching these nodes to any node in the source graph leads to a large distance. This is desirable as we apply partial OT to ignore noisy features.

However, when m reaches 0.80, important nodes (as indicated by orange arrows) start to be ignored. We aim to keep the important nodes in the coupling because they describe the main structure of the target graph. Therefore, our high-level idea in tuning m is to strike a balance between ignoring noise and preserving main features. In other words, we would like to maintain as much mass as possible in the coupling result while removing unreasonable matchings.

We assume that nodes are matched based on their Euclidean proximity. That is, matching nodes that are far apart is undesir-

able. Using partial OT, nodes that are matched faraway from each other are ignored first. By ignoring these nodes, the maximum Euclidean distance between the matched nodes in the source and the target drops drastically. Therefore, we plot the maximum Euclidean distance between the matched points w.r.t. m, referred to as the *maximum matched distance*. We use the "elbow" method to select an m where the maximum matched distance is not too high. As shown in Fig. 2 (right), for the Sinusoidal dataset, such a distance grows significantly after the elbow point at $m = 0.8625$.

We understand that such a strategy comes with some limitations. There may be multiple "elbows" in the plot, and a unified m may not work well for all pairs of Morse graphs. Without prior knowledge about the data, the choice of m may not be unique. Different m values indicate different tolerance for noises, which may all be meaningful.
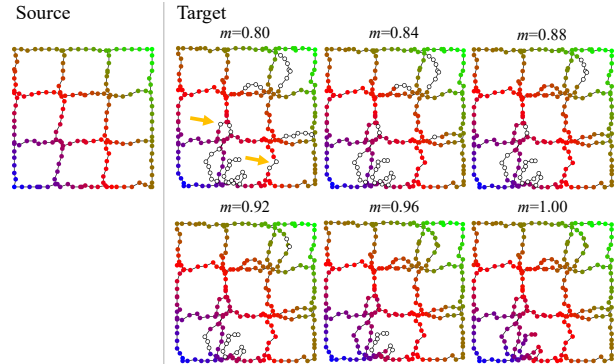

Figure 4: pFGW distance for the Sinusoidal dataset with different m values.

D RUNTIME ANALYSIS

We report the runtime in computing the OT-type distances between the source Morse graph and the target Morse graph for all real-world datasets in Tab. 1. All these distances are easy and efficient to compute. Relatively speaking, for a fixed dataset, the GW distance takes the most time. The runtime was collected on an Arch Linux system with an Intel(R) Core(TM) i7-6700K 4.00 GHz CPU with 32 GB memory.

Dataset	# of nodes	W	GW	FGW	pW	pGW	pFGW
Wind	99	0.081	0.348	0.118	0.079	0.129	0.098
Heated Cylinder	601	1.662	10.146	6.425	1.451	1.776	5.123
Navier-Stokes	130	0.184	0.278	0.130	0.118	0.146	0.178
Red Sea	171	0.183	0.531	0.250	0.166	0.283	0.286

Table 1: Runtime (in seconds) for OT-type distances between the source and the target Morse graphs across all real-world datasets.

E COMPARISON WITH EUCLIDEAN METRICS

OT-type distances can be applied to various visualization tasks. In this section, we use the HeatedCylinder dataset to demonstrate its application in detecting topological (i.e., Morse complex) changes in a time-varying scalar field. Since the topological variation between adjacent time steps is quite small, we use scalar fields at time step 800, 809, 819, 829, 839, 849, 859, 869, 879, 889 and 899 in the analysis.

There are few existing methods suitable for comparing Morse complexes. We include two simple Euclidean metrics as baseline methods for comparison. Suppose a scalar field f is uniformly sampled on a regular grid and represented as a matrix. The first metric is the Frobenius norm between the matrix representations of scalar fields, referred to as the *Euclidean scalar distance*. For the second metric, we map a Morse complex onto a binary matrix. An entry (that corresponds to a grid cell) in the matrix is 1 if it overlaps with the Morse graph; otherwise, it is 0. We then compute the Frobenius norm between the binary matrix of Morse complexes as the second metric, referred to as the *Euclidean complex distance*.
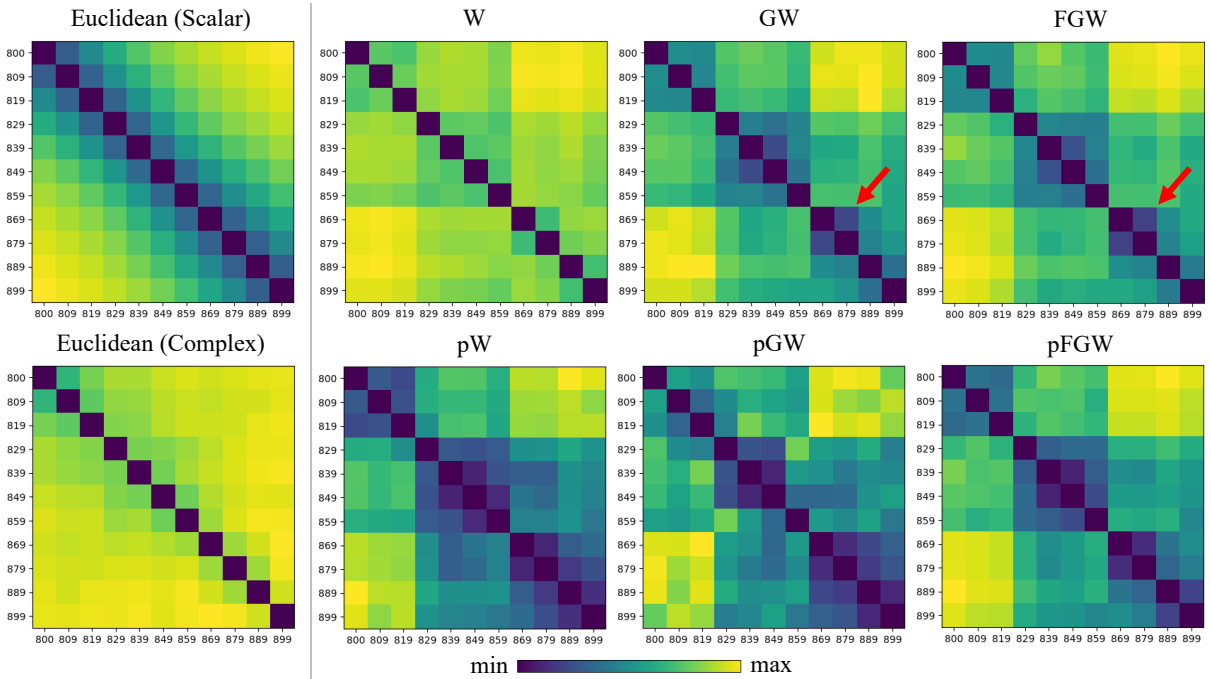
Figure 5: Pairwise distance matrix for two Euclidean metrics and six OT-type distances. Labels for both axes of matrices are time steps.
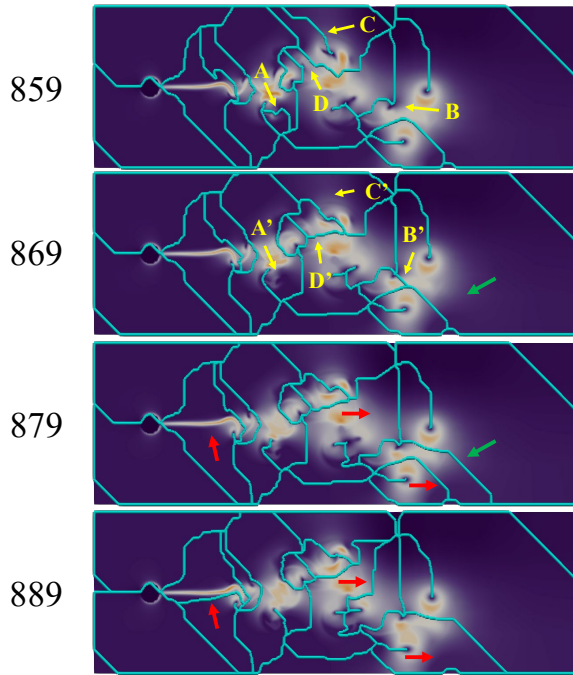

Figure 6: Scalar fields and Morse complexes in the `HeatedCylinder` dataset at time step 859, 869, 879, and 889.

Fig. 5 shows the pairwise distance matrix of each distance. Two matrices on the left are from the two Euclidean metrics. Not surprisingly, we cannot detect topological changes using these Euclidean metrics.

On there other hand, all OT-type distance matrices in Fig. 5 (right) show patterns indicating topological changes. Among the three non-partial distances, both GW and FGW distances show clear block structures (indicated by red arrows) in the matrix, indicating similarity in topology.

We further explore the data within such a block structure. Fig. 6

shows the scalar fields along with Morse complexes from time step $859 \rightarrow 889$. Changes in Morse complex at $859 \rightarrow 869$ are highlighted by yellow arrows. The edge A at time step 859 disappears at time step 869 (as indicated by A'). In the area indicated by B at time step 859, an edge appears as B' at time step 869. Similarly, $C \rightarrow C'$ indicates the disappearance of a long edge, whereas $D \rightarrow D'$ indicates a new loop enclosed by a new edge. Such structural changes are captured by the GW and FGW distances, resulting in high distances. Similarly, significant structural changes (i.e., edge appearances and disappearances) at $869 \rightarrow 879$ and at $879 \rightarrow 889$ are highlighted by green and red arrows, respectively. We observe only one green arrow at $869 \rightarrow 879$, whereas we see four yellow arrows at $859 \rightarrow 869$ and three red arrows at $879 \rightarrow 889$. This observation is consistent with the result of GW distance and FGW distance. In comparison, among four time steps from 859 to 889, the Morse complexes between time step 869 and 879 are relatively similar, whereas time step 889 is more different, time step 859 the most different.

F DEFINITION OF PARTIAL OT-TYPE DISTANCES

Following Sec. 3, we provide the formal definition of partial OT-type distances.

Partial Wasserstein distance [3]. Given $m \in [0,1]$, the *q-partial Wasserstein distance* (pW) is defined as

$$d_{q,m}^{pW}(G_1, G_2)^q = \min_{C \in \mathscr{C}_m} \sum_{i,j} d_A(a_i, b_j)^q C_{i,j}. \quad (1)$$

Partial Gromov-Wasserstein distance [3]. The *q-partial Gromov-Wasserstein distance* (pGW) is

$$d_{q,m}^{pGW}(G_1, G_2)^q = \min_{C \in \mathscr{C}_m} \sum_{i,j,k,l} |W_1(i,k) - W_2(j,l)|^q C_{i,j} C_{k,l}. \quad (2)$$

Partial Fused Gromov-Wasserstein distance [13]. For parameters $\alpha, m \in [0,1]$, the *q-partial Fused Gromov-Wasserstein distance* (pFGW) is

$$d_{q,\alpha,m}^{pFGW}(G_1, G_2)^q = \min_{C \in \mathscr{C}_m} \sum_{i,j,k,l} [(1-\alpha) d_A(a_i, b_j)^q + \alpha |W_1(i,k) - W_2(j,l)|^q] C_{i,j} C_{k,l}. \quad (3)$$

Choosing parameters appropriately, we see that all distances defined previously can be seen as special cases of the pFGW distance.

G METRIC PROPERTIES OF OT-TYPE DISTANCES

The Wasserstein and GW distances are pseudometrics on the space of A-attributed measure networks (i.e., they are symmetric, satisfy the triangle inequality, and vanish when $G_1 = G_2$), see [4]. For $\alpha \in (0, 1)$, the FGW distance is a pseudometric when $q = 1$. It is symmetric, satisfies $d_q^{FGW}(G, G) = 0$, and meets a modified triangle inequality when $q \geq 2$:

$$d_q^{FGW}(G_1, G_3) \leq 2^{1 - \frac{1}{q}} (d_q^{FGW}(G_1, G_2) + d_q^{FGW}(G_2, G_3)). \quad (4)$$

Indeed, this follows from [17, Theorem 1] (this is proved assuming that the W-functions are metrics, but the proof goes through without this assumption). The partial versions of the distances, however, fail to satisfy the triangle inequality (or any simple variant) when $m \neq 1$.

REFERENCES

[1] Computer graphics laboratory. https://cgl.ethz.ch/research/visualization/data.php. 1

[2] S. Camarri, M. Buffoni, A. Iollo, and M. V. Salvetti. Simulation of the three-dimensional flow around a square cylinder between parallel walls at moderate Reynolds numbers. In *XVII Congresso di Meccanica Teorica ed Applicata*, 2005. 1

[3] L. Chapel, M. Z. Alaya, and G. Gasso. Partial optimal tranport with applications on positive-unlabeled learning. *Advances in Neural Information Processing Systems*, 33:2903–2913, 2020. 3

[4] S. Chowdhury and F. Mémoli. The Gromov-Wasserstein distance between networks and stable network invariants. *Information and Inference: A Journal of the IMA*, 8(4):757–787, 2019. 4

[5] S. Chowdhury and T. Needham. Gromov-Wasserstein averaging in a Riemannian framework. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 842–843, 2020. 2

[6] S. Chowdhury and T. Needham. Generalized spectral clustering via Gromov-Wasserstein learning. In *International Conference on Artificial Intelligence and Statistics*, pages 712–720, 2021. 2

[7] P. Demetci, R. Santorella, B. Sandstede, W. S. Noble, and R. Singh. Scot: single-cell multi-omics alignment with optimal transport. *Journal of Computational Biology*, 29(1):3–18, 2022. 2

[8] S. Gerber, P.-T. Bremer, V. Pascucci, and R. Whitaker. Visual exploration of high dimensional scalar functions. *IEEE Transactions on Visualization and Computer Graphics*, 16:1271–1280, 2010. 1

[9] T. Günther, M. Gross, and H. Theisel. Generic objective vortices for flow visualization. *ACM Transactions on Graphics*, 36(4):141:1–141:11, 2017. 1

[10] I. Hoteit. The IEEE SciVis Contest. https://kaust-vislab.github.io/SciVis2020/, 2020. 1

[11] I. Hoteit, T. Hoar, G. Gopalakrishnan, N. Collins, J. Anderson, B. Cornuelle, A. Köhl, and P. Heimbach. A MITgcm/DART ensemble analysis and prediction system with application to the Gulf of Mexico. *Dynamics of Atmospheres and Oceans*, 63:1–23, 2013. 1

[12] International CFD Database, http://cfd.cineca.it/. 1

[13] M. Li, X. Yan, L. Yan, T. Needham, and B. Wang. Flexible and probabilistic topology tracking with partial optimal transport. arXiv preprint arXiv:2302.02895, 2023. 2, 3

[14] F. Mémoli. Gromov-Wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, 11(4):417–487, 2011. 2

[15] S. Popinet. Free computational fluid dynamics. *ClusterWorld*, 2(6), 2004. 1

[16] H. Toye, P. Zhan, G. Gopalakrishnan, A. R. Kartadikaria, H. Huang, O. Knio, and I. Hoteit. Ensemble data assimilation in the Red Sea: sensitivity to ensemble selection and atmospheric forcing. *Ocean Dynamics*, 67:915–933, 2017. 1

[17] T. Vayer, L. Chapel, R. Flamary, R. Tavenard, and N. Courty. Fused Gromov-Wasserstein distance for structured objects. *Algorithms*, 13(9):212, 2020. 4

[18] F. Vitart, C. Ardilouze, A. Bonet, A. Brookshaw, M. Chen, C. Codorean, M. Déqué, L. Ferranti, E. Fucile, M. Fuentes, et al. The subseasonal to seasonal (S2S) prediction project database. *Bulletin of the American Meteorological Society*, 98(1):163 – 173, 2017. 1

[19] W. Von Funck, T. Weinkauf, H. Theisel, and H.-P. Seidel. Smoke Surfaces: An interactive flow visualization technique inspired by real-world flow experiments. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1396–1403, 2008. 1

[20] H. Xu, D. Luo, and L. Carin. Scalable Gromov-Wasserstein learning for graph partitioning and matching. *Advances in Neural Information Processing Systems*, pages 3046–3056, 2019. 2