Hierarchical Multi-Marginal Optimal Transport for Network Alignment

Zhichen Zeng[1], **Boxin Du**[2], **Si Zhang**[3], **Yinglong Xia**[3], **Zhining Liu**[1], **Hanghang Tong**[1]
[1]University of Illinois Urbana-Champaign
[2]Amazon
[3]Meta
zhichenz@illinois.edu, boxin@amazon.com, sizhang@meta.com, yxia@meta.com, liu326@illinois.edu, htong@illinois.edu

Abstract

Finding node correspondence across networks, namely multi-network alignment, is an essential prerequisite for joint learning on multiple networks. Despite great success in aligning networks in pairs, the literature on multi-network alignment is sparse due to the exponentially growing solution space and lack of high-order discrepancy measures. To fill this gap, we propose a hierarchical multi-marginal optimal transport framework named HOT for multi-network alignment. To handle the large solution space, multiple networks are decomposed into smaller aligned clusters via the fused Gromov-Wasserstein (FGW) barycenter. To depict high-order relationships across multiple networks, the FGW distance is generalized to the multi-marginal setting, based on which networks can be aligned jointly. A fast proximal point method is further developed with guaranteed convergence to a local optimum. Extensive experiments and analysis show that our proposed HOT achieves significant improvements over the state-of-the-art in both effectiveness and scalability.

INTRODUCTION

In the era of big data, networks often originate from various domains. Joint learning on multiple networks has shown promising results in various areas including high-order recommendation (Yan et al. 2022), fraud detection (Du et al. 2021) and fact checking (Liu et al. 2021). A critical stepping-stone behind these tasks and many more is the multi-network alignment problem, which aims to find node correspondence across multiple networks.

To date, a multitude of pairwise network alignment methods have been developed based on the consistency principle (Singh, Xu, and Berger 2008; Koutra, Tong, and Lubensky 2013; Zhang and Tong 2016), node embedding (Li et al. 2019; Chu et al. 2019; Zhang et al. 2021), and optimal transport (OT) (Maretic et al. 2019, 2022; Chen et al. 2020; Zeng et al. 2023a) with superior performance, but this is not the case for the multi-network setting due to two fundamental challenges. First (*discrepancy measure*), most existing pairwise methods essentially optimize the pairwise discrepancy (e.g., Frobenius norm (Zhang and Tong 2016), contrastive loss (Chu et al. 2019), and Wasserstein distance (Maretic et al. 2020)) between one network and its aligned counterpart, but a similar discrepancy measure for multi-network is lacking. Second (*algorithm*), even equipped with a proper discrepancy measure, an efficient algorithm is demanded to handle the significantly larger solution space of multi-network alignment, compared with its pairwise counterpart.

Contributions. In this paper, we propose a novel method named HOT to address the above challenges from the view of multi-marginal optimal transport (MOT) (Pass 2015). To jointly measure the discrepancy between multiple networks, the fused Gromov-Wasserstein (FGW) distance is generalized to the multi-marginal setting, whose by-product, the optimal coupling tensor, naturally serves as the alignment between networks. To handle the large solution space, the problem is decomposed into significantly smaller cluster-level and node-level alignment subproblems. Specifically, the cluster-level alignment for multiple networks is obtained based on the FGW barycenter. On top of that, the multi-marginal FGW (MFGW) distance, together with a position-aware cost tensor generated based on the unified random walk with restart (RWR), is adopted for the node-level alignment. To achieve fast solutions, we propose a proximal point method with guaranteed convergence. Extensive experiments show that HOT outperforms the best competitor by at least 12.0% on plain networks in terms of high-order Hits@10, with up to 360× speedup in time complexity and 1000× reduction in memory cost compared with the non-hierarchical solution.

The rest of the paper is organized as follows. We first introduces the preliminaries and problem definitions, and then formulate the optimization problem. We further present and analyze the proposed algorithm, followed by extensive experiment results. Related work and conclusions are presented afterwards.

PROBLEM DEFINITION

Notations

We use bold uppercase letters for matrices (e.g., $\mathbf{A}$), bold lowercase letters for vectors (e.g., $\mathbf{s}$), calligraphic letters for sets (e.g., $\mathcal{C}$), bold calligraphic letters for tensors (e.g., $\boldsymbol{\mathcal{C}}$), and lowercase letters for scalars (e.g., α). The element (i, j) of a matrix $\mathbf{A}$ is denoted as $\mathbf{A}(i, j)$, and the element $(i_1, i_2, \ldots, i_K)$ of a tensor $\boldsymbol{\mathcal{C}}$ is denoted as $\boldsymbol{\mathcal{C}}(i_1, i_2, \ldots, i_K)$. The transpose of $\mathbf{A}$ is denoted by the superscript $^\intercal$ (e.g., $\mathbf{A}^\intercal$). We use $\Pi(\boldsymbol{\mu}, \boldsymbol{\nu})$ to denote the probabilistic coupling between

$\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, and $\Delta_n = \{\boldsymbol{\mu} \in \mathbb{R}_n^+ | \sum_{i=1}^n \boldsymbol{\mu}(i) = 1\}$ to denote a probability simplex with n bins.

For mathematical operations, we use $\odot$ for Hadmard product and $\otimes$ for outer product. We define $\mathcal{P}_k(\boldsymbol{\mathcal{C}}) = \sum_{\{i_1,\ldots,i_K\}\setminus\{i_k\}} \boldsymbol{\mathcal{C}}(i_1,\ldots,i_K)$ as the marginal sum of tensor $\boldsymbol{\mathcal{C}}$ of the k-th dimension.

An attributed graph is denoted as $\mathcal{G} = \{\mathbf{A}, \mathbf{X}\}$, where $\mathbf{A}$ is the adjacency matrix and $\mathbf{X}$ is the node attribute matrix. We use n_i and m_i to denote the number of nodes and edges in $\mathcal{G}_i$, respectively. Graph indices are indicated by subscripts (e.g., $\mathcal{G}_i$) and cluster indices are indicated by superscripts (e.g., $\mathcal{C}^j$). For a given graph $\mathcal{G}_k$, the i_k-th node is denoted as v_{i_k}, and the j-th cluster is denoted as $\mathcal{C}_k^j$.

Following a common practice in OT-based graph applications (Titouan et al. 2019), an attributed graph can be represented by a probability measure supported on the product space of node attribute and structure, i.e., $\boldsymbol{\mu} = \sum_{i=1}^n \mathbf{h}(i)\delta_{v_i,\mathbf{X}(v_i)}$, where $\mathbf{h} \in \Delta_n$ is a histogram representing the node weight of $v_i \in \mathcal{G}$.

Multi-marginal Optimal Transport

The fused Gromov-Wasserstein (FGW) distance is powerful in processing geometric data by exploring node attributes and graph structure, which is defined as (Titouan et al. 2019):

Definition 1. *Fused Gromov-Wasserstein (FGW) distance. Given two graphs $\mathcal{G}_1 = \{\mathbf{A}_1, \mathbf{X}_1\}, \mathcal{G}_2 = \{\mathbf{A}_2, \mathbf{X}_2\}$ with their probability measures $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$ and intra-cost matrices $\mathbf{C}_1, \mathbf{C}_2$ measuring within-graph node relationships, and a cross-cost matrix $\mathbf{C}_{\text{cross}}$ measuring cross-graph node relationships, the FGW distance $\text{FGW}_{q,\alpha}(\mathcal{G}_1, \mathcal{G}_2)$ is defined as*

$$\min_{\mathbf{S}\in\Pi(\boldsymbol{\mu}_1,\boldsymbol{\mu}_2)} (1-\alpha) \sum_{v_1\in\mathcal{G}_1, u_1\in\mathcal{G}_2} \mathbf{C}_{\text{cross}}^q(v_1,u_1)\mathbf{S}(v_1,u_1) + \alpha \sum_{\substack{v_1,v_2\in\mathcal{G}_1 \\ u_1,u_2\in\mathcal{G}_2}} |\mathbf{C}_1(v_1,v_2) - \mathbf{C}_2(u_1,u_2)|^q \mathbf{S}(v_1,u_1)\mathbf{S}(v_2,u_2). \quad (1)$$

The hyperparameter q in Eq. (1) is the order of the FGW distance, and we consider $q = 2$ for faster computation throughout this paper (Peyré, Cuturi, and Solomon 2016). However, existing FGW distance is only applicable in two-sided OT problems. Based on Definition 1, the FGW distance is generalized to the multi-marginal OT setting as follows:

Definition 2. *Multi-marginal Fused Gromov-Wasserstein (MFGW) distance (Beier, Beinert, and Steidl 2022). Given K graphs $\mathcal{G}_1,\ldots,\mathcal{G}_K$ with their probabilistic representations $\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K$, a cross-cost tensor $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{n_1\times\cdots\times n_K}$ measuring cross-graph node distances based on node attributes, and K intra-cost matrices $\mathbf{C}_k \in \mathbb{R}^{n_k\times n_k}, \forall k = 1,\ldots,K$ measuring intra-graph node similarity for $\mathcal{G}_k$ based on graph structure. The q-MFGW distance $\text{MFGW}_{q,\alpha}(\mathcal{G}_1,\ldots,\mathcal{G}_K)$ is defined as:*

$$\min_{\boldsymbol{\mathcal{S}}\in\Pi(\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_K)} (1-\alpha) \sum_{v_1,\ldots,v_K} \boldsymbol{\mathcal{C}}(v_1,\ldots,v_K)^q \boldsymbol{\mathcal{S}}(v_1,\ldots,v_K) + \alpha \sum_{\substack{1\le j,k\le K \\ v_1,\ldots,v_K \\ v_1',\ldots,v_K'}} |\mathbf{C}_j(v_j,v_j') - \mathbf{C}_k(v_k,v_k')|^q \boldsymbol{\mathcal{S}}(v_1,\ldots,v_K)\boldsymbol{\mathcal{S}}(v_1',\ldots,v_K') \quad (2)$$

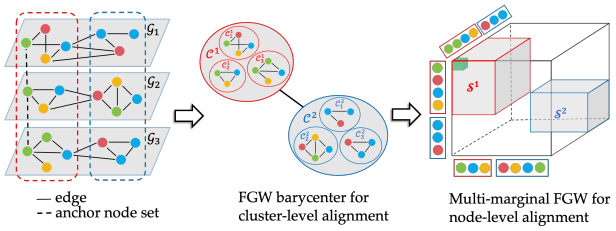


Figure 1: An overview of HOT. Left: three input networks, where three green nodes connected by the black dash line form an anchor node set. Middle: FGW barycenter co-clusters three graphs into two clusters. Right: the node alignment tensor with blocks $\boldsymbol{\mathcal{S}}^1$ for cluster $\mathcal{C}^1$ and $\boldsymbol{\mathcal{S}}^2$ for cluster $\mathcal{C}^2$.

Intuitively, the first summation is the Wasserstein term measuring the joint distance for K graphs in terms of node attributes. The second summation is the Gromov-Wasserstein term measuring the structural difference among all node pairs in K graphs weighted by the optimal coupling tensor $\boldsymbol{\mathcal{S}}$.

Hierarchical Multi-network Alignment

Problem 1. *Hierarchical multi-network alignment.*
Given: *(1) K attributed networks $\mathcal{G}_i = \{\mathbf{A}_i, \mathbf{X}_i\}$, and (2) a set of anchor node sets $\mathcal{L}$ indicating which nodes are aligned a priori.*
Output: *(1) cluster-level alignment sets $\mathcal{C}^j = \bigcup_{i=1}^K \mathcal{C}_i^j$ for $j = 1,\ldots,M$, where M is the number of clusters and $\mathcal{C}_i^j$ is the set of nodes from $\mathcal{G}_i$ that are clustered to the j-th cluster, and (2) node-level alignment tensors $\boldsymbol{\mathcal{S}}^j$ for $\mathcal{C}^j$, whose entry indicates how likely nodes are aligned.*

An illustrative example is shown in Figure 1. Given the anchor node set $\mathcal{L}$, the 1^{st} cluster-level alignment $\mathcal{C}^1$ (red circle in the middle figure) consists of clusters $\mathcal{C}_1^1$, $\mathcal{C}_2^1$ and $\mathcal{C}_3^1$, and corresponding $\boldsymbol{\mathcal{S}}^1$ indicates alignments among nodes in $\mathcal{C}_1^1$, $\mathcal{C}_2^1$ and $\mathcal{C}_3^1$. Note that Problem 1 is a generalized version of single-level pairwise network alignment. For example, when $K = 2$, the problem degenerates to the hierarchical pairwise alignment problem (Xu, Luo, and Carin 2019; Zhang et al. 2019). When $M = 1$, the problem degenerates to the single-level multi-network alignment problem (Chu et al. 2019).

OPTIMIZATION FORMULATION

In this section, we present our hierarchical MOT-based multi-network alignment framework. A position-aware cost tensor is first developed to depict high-order relationships across networks. Then the multi-network alignment problem is formulated as a hierarchical MOT problem, including cluster-level alignmend based on FGW barycenter and node-level alignment based on MFGW distance.

Position-Aware Cost Tensor

Modeling node relationship across multiple networks is essential for multi-network alignment. Consistency-based methods (Du, Liu, and Tong 2021; Li et al. 2021; Zhang and Tong 2016) model node relationships by the Kronecker product graph, the size of which becomes intractable for large networks. Embedding-based methods (Heimann et al. 2018;

Zhang et al. 2020, 2021) generate embedding spaces for different network pairs but suffer from the space disparity issue.

To overcome the above limitations, we adopt the unified RWR to generate position-aware node embeddings in a unified space (Yan, Zhang, and Tong 2021; Yan et al. 2024). The idea is to treat nodes in an anchor node set as one identical landmark in the embedding space and construct a unified space by encoding positional information with respect to (w.r.t.) same landmarks. Formally speaking, given the p-th anchor node set $\{l_{1_p}, \ldots, l_{K_p}\} \in \mathcal{L}$ where l_{i_p} is the anchor node from $\mathcal{G}_i$, the RWR score vector $\mathbf{r}_{i_p} \in \mathbb{R}^{n_i}$ depicting the relative positions of nodes from $\mathcal{G}_i$ w.r.t. l_{i_p} is computed by (Tong, Faloutsos, and Pan 2006)

$$\mathbf{r}_{i_p} = (1-\beta)\mathbf{W}_i\mathbf{r}_{i_p} + \beta\mathbf{e}_{i_p}, \tag{3}$$

where β is the restart probability, $\mathbf{W}_i = (\mathbf{D}_i^{-1}\mathbf{A}_i)^\intercal$ is the transpose of the row normalized matrix of $\mathbf{A}_i$, and $\mathbf{e}_{i_p}$ is an n_i-dimensional one-hot vector with $\mathbf{e}_{i_p}(l_{i_p}) = 1$. The final positional embedding is the concatenation of the RWR scores w.r.t. different anchor node sets in $\mathcal{L}$, i.e., $\mathbf{R}_i = [\mathbf{r}_{i_1} \| \ldots \| \mathbf{r}_{i_{|\mathcal{L}|}}] \in \mathbb{R}^{n_i \times |\mathcal{L}|}$.

When node attributes are available, we use the concatenation of node attribute and positional embedding as the node embedding, i.e., $\mathbf{Z}_i = [\mathbf{X}_i \| \mathbf{R}_i]$ for $\mathcal{G}_i$. Otherwise, we simply use $\mathbf{R}_i$ as the node embedding, i.e., $\mathbf{Z}_i = \mathbf{R}_i$. Given a set of node embeddings $\{\mathbf{Z}_1, \ldots, \mathbf{Z}_K\}$, the position-aware cost tensor $\boldsymbol{\mathcal{C}}$ is computed by the total sum of all pairwise node distances as follows (Alaux et al. 2019):

$$\boldsymbol{\mathcal{C}}(v_1, \ldots, v_K) = \sum_{1 \le j,k \le K} \|\mathbf{Z}_j(v_j) - \mathbf{Z}_k(v_k)\|_2. \tag{4}$$

FGW-based Cluster-level Alignment

Hierarchical structures are ubiquitous in real-world networks, and exploring such cluster structures can benefit the multi-network alignment task in both effectiveness and scalability (Zhang et al. 2019; Jing et al. 2023; Liu et al. 2019). For example, as shown in Figure 1, if cluster-level alignments are known, we can dramatically shrink the solution space by only considering nodes in the aligned clusters for node-level alignments. To obtain high-quality cluster-level alignments, we follow a similar approach as (Xu, Luo, and Carin 2019) based on the FGW barycenter (Titouan et al. 2019).

Given K networks $\mathcal{G}_i = \{\mathbf{A}_i, \mathbf{X}_i\}$ and their probability measures $\boldsymbol{\mu}_i$, the FGW barycenter $\mathcal{G}_b = \{\mathbf{A}_b, \mathbf{X}_b\}$ serves as a consensus graph that is close to all given graphs in terms of the FGW distance. Regarding each node in $\mathcal{G}_b$ as the barycenter of one cluster, nodes transported to the same barycenter form a cluster-level alignment. Specifically, we adopt the L_2 norm between node attributes as the cross-cost matrices, i.e., $\mathbf{C}_{\text{cross}_i}(v,u) = \|\mathbf{X}_i(v) - \mathbf{X}_b(u)\|_2, \forall v \in \mathcal{G}_i, u \in \mathcal{G}_b$, to depict node relationships between $\mathcal{G}_i$ and $\mathcal{G}_b$. The FGW-based cluster-level alignment problem is formulated as

$$\arg\min_{\mathbf{A}_b, \mathbf{X}_b} \sum_{i=1}^{K} \text{FGW}_{2,\alpha}(\mathbf{C}_{\text{cross}_i}, \mathbf{A}_i, \mathbf{A}_b, \boldsymbol{\mu}_i, \boldsymbol{\mu}_b). \tag{5}$$

By exploiting the OT coupling $\mathbf{S}_i$ between $\mathcal{G}_i$ and barycenter $\mathcal{G}_b$, nodes $v_i \in \mathcal{G}_i$ are determinisitically assigned to the barycenter node $b_j \in \mathcal{G}_b$ such that $b_j = \arg\max_{b \in \mathcal{G}_b} \mathbf{S}_i(v_i, b)$. Note that these barycenter nodes b_j serves as "references" connecting nodes v_i in cluster $\mathcal{C}_i^j$ in different graph $\mathcal{G}_i$, hence providing a cluster-level alignment $\mathcal{C}^j = \bigcup_{i=1}^K \mathcal{C}_i^j$. An illustrative example is given by the middle subfigure of Figure 1.

MFGW-based Node-level Alignment

The MFGW distance in Definition 2 provides a joint distance measure for multiple networks given their attributes and structure, and the optimal coupling $\boldsymbol{\mathcal{S}}$, as a by-product of the MFGW distance, indicates the high-order node alignments across networks.

To make the computation more tractable, we first propose a tensor form MFGW distance as follows

Proposition 1. *The MFGW distance in Eq.* (2) *with* $q = 2$ *can be formulated into a tensor form as:*

$$\min_{\boldsymbol{\mathcal{S}} \in \Pi(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)} \langle (1-\alpha)\boldsymbol{\mathcal{C}} + \alpha\boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{S}} \rangle, \tag{6}$$

where $\boldsymbol{\mathcal{L}}(v_1, ..., v_K) = (K-1)\sum_{j=1}^K \mathbf{C}_j(v_j, \cdot)^2 \mathcal{P}_j(\boldsymbol{\mathcal{S}}) - 2\sum_{1 \le j < k \le K} \mathbf{C}_j(v_j, \cdot)\mathcal{P}_{j,k}(\boldsymbol{\mathcal{S}})\mathbf{C}_k(v_k, \cdot)^\intercal$.

Directly applying the MFGW on node alignments still leads to intractable time and space complexities. To overcome this issue, we achieve an exponential reduction in both complexities by only considering node alignments inside the aligned clusters $\mathcal{C}^j$, which decomposes the original problem of size $\mathcal{O}(n^K)$ into M independent in-cluster node-level alignment subproblems, each with size $\mathcal{O}\left(\left(\frac{n}{M}\right)^K\right)$. Following a common practice (Titouan et al. 2019), we represent clusters $\mathcal{C}_i^j \in \mathcal{C}^j$ as discrete uniform distributions $\boldsymbol{\mu}_i^j = \mathbf{1}/|\mathcal{C}_i^j|$ supported on its nodes. Together with the position-aware cost tensor $\boldsymbol{\mathcal{C}}^j$ in Eq. (4) and the intra-cluster adjacency matrices $\mathbf{A}_1^j, \ldots, \mathbf{A}_2^j$ describing intra-cluster node connectivity, the node-level alignment subproblem is formulated as the following MFGW problem:

$$\boldsymbol{\mathcal{S}}^j = \arg\min_{\boldsymbol{\mathcal{S}} \in \Pi(\boldsymbol{\mu}_1^j, \ldots \boldsymbol{\mu}_K^j)} \langle (1-\alpha)\boldsymbol{\mathcal{C}}^j + \alpha\boldsymbol{\mathcal{L}}^j, \boldsymbol{\mathcal{S}} \rangle, \forall j = 1, \ldots, M, \tag{7}$$

where $\boldsymbol{\mathcal{S}}^j$ is the node-level alignment tensor for $\mathcal{C}^j$.

ALGORITHM AND ANALYSIS

In this section, we present and analyze our optimization algorithm HOT. We first adopt the block coordinate descent (BCD) method to solve the FGW-based cluster-level alignment. Afterward, the MFGW-based node-level alignment is solved by the proximal point method to a local optimum. Relevant analyses of the proposed HOT are carried out thereafter.

Optimization Algorithm

FGW-based cluster-level alignment in Eq. (5) is a non-convex multivariate optimization problem and can be efficiently solved by the BCD algorithm (Ferradans et al. 2014). Specifically, the objective is minimized w.r.t. $\mathbf{S}_i$, $\mathbf{A}_b$ and $\mathbf{X}_b$ iteratively. For the t-th iteration, the minimization w.r.t. three variables are calculated as follows.

First, fixing $\mathbf{A}_b$ and $\mathbf{X}_b$, the optimization w.r.t. $\mathbf{S}_i$ is formulated as

$$\begin{aligned}\mathbf{S}_i^{(t+1)} &= \sum_{j=1}^{K} \min_{\mathbf{S}_j \in \Pi(\boldsymbol{\mu}_j, \boldsymbol{\mu}_b)} \langle (1-\alpha)\mathbf{C}_{\text{cross}_j}^{(t)} + \alpha \mathbf{L}_j^{(t)}, \mathbf{S}_j \rangle \\ &= \min_{\mathbf{S}_i \in \Pi(\boldsymbol{\mu}_i, \boldsymbol{\mu}_b)} \langle (1-\alpha)\mathbf{C}_{\text{cross}_i}^{(t)} + \alpha \mathbf{L}_i^{(t)}, \mathbf{S}_i \rangle.\end{aligned} \tag{8}$$

The last equation is due to the fact that $\mathbf{S}_i^{(t)}$ are decoupled from each other, so it is equivalent to minimizing K FGW distances independently. Note that the optimization problem in Eq. (8) is a special case (i.e., two-sided OT setting) of the MFGW problem in Definition 2, and can be efficiently solved by the proximal point method introduced later in this section.

Second, fixing $\mathbf{S}_i$ and $\mathbf{X}_b$, the optimal value for the adjacency matrix $\mathbf{A}_b$ of $\mathcal{G}_b$ can be computed by the first-order optimality condition as (Peyré, Cuturi, and Solomon 2016)

$$\mathbf{A}_b^{(t+1)} = \frac{\mathbf{1}_{M\times M}}{\boldsymbol{\mu}_b \boldsymbol{\mu}_b^\top} \sum_{i=1}^{K} \left(\mathbf{S}_i^{(t+1)^\top} \mathbf{A}_i \mathbf{S}_i^{(t+1)} \right). \tag{9}$$

Third, fixing $\mathbf{S}_i$ and $\mathbf{A}_b$, the objective function is quadratic w.r.t. the node attribute matrix $\mathbf{X}_b$, whose optimal value can be efficiently computed as (Cuturi and Doucet 2014)

$$\mathbf{X}_b^{(t+1)} = \sum_{i=1}^{K} \left(\text{diag}\left(\frac{\mathbf{1}_M}{\boldsymbol{\mu}_b} \right) \mathbf{S}_i^{(t+1)^\top} \mathbf{X}_i \right). \tag{10}$$

By iteratively applying Eqs. (8)-(10), the algorithm converges to the local optimal barycenter (Titouan et al. 2019).

MFGW-based node-level alignment. In order to handle the non-convex objective function in Eq. (7), we generalize the proximal point method (Xu et al. 2019) to the multi-marginal setting with guaranteed convergence to a local optimum. The key idea is to decompose the non-convex problem into a series of convex subproblems regularized by the proximal operator. We adopt the KL divergence as the proximal operator, i.e., $\text{KL}(\boldsymbol{\mathcal{S}} \| \boldsymbol{\mathcal{S}}^{(t)})$, to regularize the distance between two successive solutions, and the resulting problem corresponds to a regularized MOT problem as follows:

$$\begin{aligned}\boldsymbol{\mathcal{S}}^{(t+1)} &= \underset{\boldsymbol{\mathcal{S}} \in \Pi(\boldsymbol{\mu}_1^j, \ldots, \boldsymbol{\mu}_K^j)}{\arg\min} \langle (1-\alpha)\boldsymbol{\mathcal{C}} + \alpha \boldsymbol{\mathcal{L}}^{(t)}, \boldsymbol{\mathcal{S}} \rangle + \lambda \text{KL}(\boldsymbol{\mathcal{S}} \| \boldsymbol{\mathcal{S}}^{(t)}) \\ &= \underset{\boldsymbol{\mathcal{S}} \in \Pi(\boldsymbol{\mu}_1^j, \ldots, \boldsymbol{\mu}_K^j)}{\arg\min} \langle \boldsymbol{\mathcal{Q}}^{(t)}, \boldsymbol{\mathcal{S}} \rangle + \lambda \langle \boldsymbol{\mathcal{S}}, \log \boldsymbol{\mathcal{S}} \rangle,\end{aligned} \tag{11}$$

where $\boldsymbol{\mathcal{Q}}^{(t)} = (1-\alpha)\boldsymbol{\mathcal{C}} + \alpha \boldsymbol{\mathcal{L}}^{(t)} - \lambda \log \boldsymbol{\mathcal{S}}^{(t)}$ is fixed when optimizing $\boldsymbol{\mathcal{S}}$; hence, the resulting problem corresponds to an entropic regularized OT problem with modified cost tensor $\boldsymbol{\mathcal{Q}}^{(t)}$ and can be efficiently solved by the Sinkhorn algorithm (Cuturi 2013).

Specifically, with initial scaling vectors $\mathbf{u}_i^{(0)}$, the algorithm iteratively updates scaling vectors by

$$\mathbf{u}_i^{(l+1)} = \frac{\mathbf{u}_i^{(l)} \odot \boldsymbol{\mu}_i^j}{\mathcal{P}_i \left[\exp(-\frac{\boldsymbol{\mathcal{Q}}^{(t)}}{\lambda}) \odot \bigotimes_{i=1}^{K} \mathbf{u}_i^{(l)} \right]}. \tag{12}$$

After L inner iterations of Eq. (12), the final solution $\boldsymbol{\mathcal{S}}^{(t+1)}$ can be computed as

$$\boldsymbol{\mathcal{S}}^{(t+1)} = \exp(-\frac{\boldsymbol{\mathcal{Q}}^{(t)}}{\lambda}) \odot \bigotimes_{i=1}^{K} \mathbf{u}_i^{(L)}. \tag{13}$$

As we will show in the following analysis, by iteratively applying Eqs. (11)-(13), the solution sequence given by the proposed proximal point method converges to a local optimum of the MFGW distance.

Theoretical Analysis

Without loss of generality, we assume that networks share a comparable size, each with $\mathcal{O}(n)$ nodes and $\mathcal{O}(m)$ edges. For brevity, we denote the average cluster size as $\overline{n} = \frac{n}{M}$.

Complexity analysis. We first provide a complexity analysis of the proposed HOT as follows

Proposition 2. *With K graphs, M clusters, and T proximal point iterations, the space complexity of* HOT *is $\mathcal{O}(M\overline{n}^K)$, and the time complexity is $\mathcal{O}(TKM(n^2 + K\overline{n}^K))$.*

For space complexity, the overall $\mathcal{O}(M\overline{n}^K)$ achieves an exponential reduction of space in terms of the number of graphs K compared to $\mathcal{O}(n^K)$ given by the straightforward method, which finds the full node-level alignment tensor without cluster-level alignment. For time complexity, the first term $\mathcal{O}(TKMn^2)$ corresponds to the cluster-level alignment, and the second term $\mathcal{O}(TK^2M\overline{n}^K)$ accounts for the node-level alignment. For cases where $\mathcal{O}(\overline{n}) < \mathcal{O}\left(\sqrt[K]{n^2/K} \right)$, the time complexity is determined by the cluster-level alignment and can be approximated by $\mathcal{O}(TKMn^2)$, which is quadratic w.r.t. n and linear w.r.t. K. Otherwise, the time complexity mostly lies in the node-level alignment and can be approximated by $\mathcal{O}(TK^2M\overline{n}^K)$, which is polynomial w.r.t. $\overline{n}$ and exponential w.r.t. K. Since that $\overline{n} = \frac{n}{M}$, we achieve an exponential reduction of time in terms of the number of graphs K compared to $\mathcal{O}(TK^2n^K)$ of the straightforward method.

Optimality and convergence. First, for the position-aware cost tensor in Eqs. (3) and (4), the computation has guaranteed convergence via the fixed point method as the eigenvalues of $\mathbf{W}_i$ lie in [-1,1]. Second, for the FGW barycenter computation, the solution sequence given by the BCD method converges to a stationary point (Titouan et al. 2019). Third, for the MFGW computation, we have the following proposition stating that the proximal point method converges to a local optimum of the MFGW problem.

Proposition 3. *The solution sequence $\boldsymbol{\mathcal{S}}^{(t)}$ given by the proximal point method converges to a stationary point of the MFGW problem in Definition 2.*

The general idea is to show the regularized objective is an upper bound of the original objective and further take advantage of the convergence theorem of the successive upper-bound minimization method (Razaviyayn, Hong, and Luo 2013; Xu et al. 2019). Therefore, the proposed HOT is guaranteed to converge to the local optimum.

Connection with pairwise FGW distance. Besides, we reveal the close connection between FGW and MFGW distance. When adopting the square loss, i.e., $q = 2$, the MFGW

distance is lower bounded by the sum of FGW distances between all possible pairs. In other words, the joint distance between multiple networks is likely to be underestimated by the pairwise FGW distance.

Theorem 1. *Given K graphs $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_K$, the MFGW distance is lower bounded by the sum of all pairwise FGW distances, that is:*

$$\sum_{1\leq j<k\leq K} \text{FGW}_{2,\alpha}(\mathcal{G}_j, \mathcal{G}_k) \leq \text{MFGW}_{2,\alpha}(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_K).$$

EXPERIMENTS

We evaluate the proposed HOT from the following aspects:

- Q1. How effective is the proposed HOT?
- Q2. How scalable is the proposed HOT?
- Q3. How is the convergence of the proposed HOT?
- Q3. How robust is the proposed HOT to hyperparameters?

Datasets. Our method is evaluated on both plain networks, including Douban, ER and DBLP, and attributed networks, including ACM(A) and DBLP(A). To mitigate the effect of data split, we randomly split the datatsets into 10 folds, using 1 fold (i.e., 10%) for training and the rest 9 folds for testing. We report the mean and standard deviation of the alignment results with different training/test splits[1].

Baseline methods. The proposed HOT is compared with a variety of baseline methods, including (1) consistency-based methods: IsoRank (Singh, Xu, and Berger 2008), FINAL (Zhang and Tong 2016), MOANA (Zhang et al. 2019), and SYTE (Du, Liu, and Tong 2021), (2) embedding-based methods: CrossMNA (Chu et al. 2019), NetTrans (Zhang et al. 2020), NeXtAlign (Zhang et al. 2021), and Grad-Align (Park et al. 2022), and (3) OT-based methods: GW (Mémoli 2011), FGW (Titouan et al. 2019), Low-rank OT (LOT) (Scetbon, Cuturi, and Peyré 2021), S-GWL (Xu, Luo, and Carin 2019), and WAlign (Gao, Huang, and Li 2021). For pairwise alignment methods, we run them on each pair of networks and integrate the alignment matrices by multiplication (e.g., for networks $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3$, the alignment tensor is obtained by $\boldsymbol{\mathcal{S}}(x, y, z) = \mathbf{S}_{\mathcal{G}_1,\mathcal{G}_2}(x, y)\mathbf{S}_{\mathcal{G}_1,\mathcal{G}_3}(x, z)\mathbf{S}_{\mathcal{G}_2,\mathcal{G}_3}(y, z)$).

Parameter settings. In our experiments, we adopt a consistent parameter setting with $\lambda = 10^{-3}$, $\alpha = 0.5$, and $\beta = 0.15$. For number of clusters, we set $M = \lceil \frac{n}{50} \rceil$ for all datasets.

Metrics. We evaluate the effectiveness in terms of pairwise Hits@K (PH@K), high-order Hits@K (HH@K) (Du, Liu, and Tong 2021) and Mean Reciprocal Rate (MRR). Given a test node $x_1 \in \mathcal{G}_1$, if any corresponded node $x_i \in \mathcal{G}_i$ exists in the top-K most similar node sets, it is regarded as a pairwise hit. Only if the whole corresponded node set $\{x_i\}$ appears in the top-K most similar node sets, it is regarded as a high-order hit. For a test dataset with n node sets, the Hits@K is computed by Hits@$K = \frac{\#\text{ of hits}}{n}$, and MRR is computed by the average of the inverse of high-order alignment ranking MRR $= \frac{1}{n}\sum_{i=1}^{n} \frac{1}{\text{rank}(\{x_i\})}$

[1]Code and datasets are available at https://github.com/zhichenz98/HOT-AAAI24

Effectiveness Results

The alignment results on plain and attributed networks are shown in Figures 2 and 3 respectively. For better visualization, we use "$\times$" for consistency-based, "$\bullet$" for embedding-based and "$\star$" for OT-based methods.

It is shown that HOT outperforms all baselines in most cases with more significant outperformance on

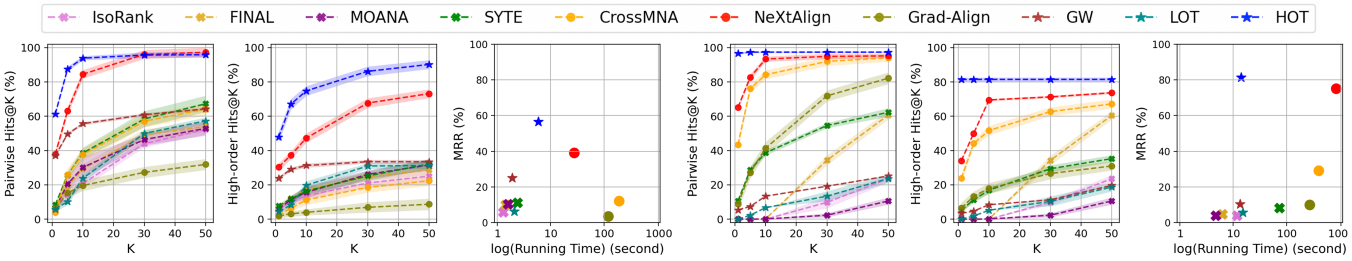
- plain networks than attributed networks, thanks to the position-aware cost tensor and the MFGW distance addressing the topological relationship across multiple networks jointly, which is particularly important to plain networks where node attributes are unavailable.
- high-order metrics than pairwise metrics, as the proposed HOT aligns networks jointly, whereas many baselines align networks in pairs, resulting in incompatible alignment scores and node embeddings in disparate spaces.
- harder metrics (e.g., Hits@1) than softer metrics (e.g., Hits@50). This is due to the exponential term in Eq. (8) that provides more deterministic and noise-reduced alignments (Mena et al. 2018; Zeng et al. 2023a), compared with the uncertain alignments given by many baselines.

Compared with consistency-based methods, HOT outperforms the best competitor by at least 55.3% in PH@10, 51.9% in HH@10, and 42.9% in MRR on plain networks. On attributed networks, HOT surpasses the best competitor[2] at least 57.3% in PH@10, 53.3% in HH@10, and 68.2% in MRR. The limited performance of consistency-based methods owes to the fact that the consistency principle only enforces the consistency between aligned node pairs (Zhang and Tong 2016) and fail to model the high-order node relationships jointly. Besides, HOT achieves comparable running time as consistency-based methods when aligning small networks (e.g., Douban-230 and ER-500) and faster speed when aligning large networks (e.g., ACM(A)-1000 and DBLP(A)-1000).

In comparison to embedding-based methods, HOT outperforms the best competitor by at least 4.0% in PH@10, 12.0% in HH@10, and 1.2% in MRR on plain networks. On attributed networks, HOT surpasses the best competitor by at least 3.1% in PH@10, 1.0% in HH@10, and 2.8% in MRR. While the best competitor NEXTALIGN (Zhang et al. 2021) achieves comparable PH@K, HOT significantly outperforms it in HH@K and MRR. This is because embedding-based methods basically optimize a ranking-based loss addressing pairwise distances, while the joint high-order relationships are largely ignored.

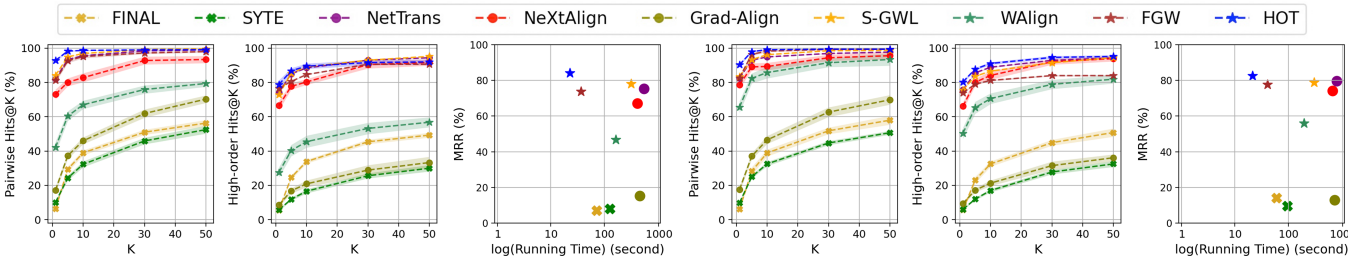
For OT-based methods, empirical evaluation shows that HOT outperforms the best competitor by at least 35.4% in PH@10, 15.8% in HH@10, and 6.3% in MRR. On attributed networks, HOT outperforms the best OT-based competitor by at least 0.2% in PH@10, 0.4% in HH@10, and 1.4% in MRR. Although LOT adopts a similar idea as HOT that explores the low rank structure in the pairwise alignment matrix, i.e., cluster structure in graphs, the low rank structure is inconsistent for different network pairs, hence may fail to be generalized to the multi-network setting.

[2]Slight deviations from results in (Du, Liu, and Tong 2021) may exist due to different ways to categorize node attributes.



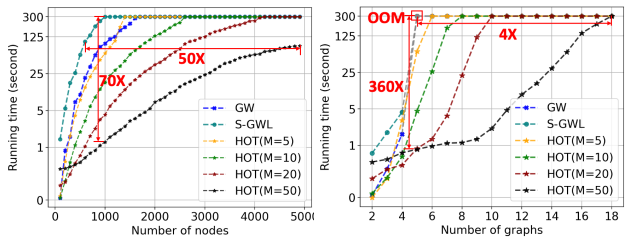
(a) Douban-230 (b) ER-500

Figure 2: Alignment results on plain networks: (a) Douban-230; (b) ER-500.



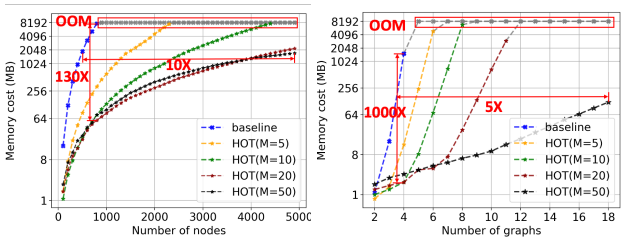
(a) ACM(A)-1000 (b) DBLP(A)-1000

Figure 3: Alignment results on attributed networks: (a) ACM(A)-1000; (b) DBLP(A)-1000.



(a) (b)

Figure 4: Experiments on time complexity w.r.t. (a) number of nodes, and (b) number of graphs: grey points indicate out-of-memory. Note that the running time is in the log scale.



(a) (b)

Figure 5: Experiments on space complexity w.r.t. (a) number of nodes, and (b) number of graphs: grey points indicate out-of-memory. Note that the memory cost is in the log scale.

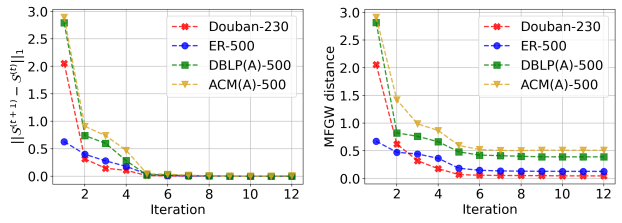
Scalability Results

We study the scalability of the proposed HOT. In general, we evaluate the scalability w.r.t. the number of nodes n by aligning three ER graphs with different sizes, and the scalability w.r.t. the number of graphs K by aligning multiple ER graphs with 100 nodes. The time complexity and space complexity results are shown in Figure 4 and Figure 5 respectively. We terminate the program if it can not finish in 300 seconds or exceeds the memory capacity (8GB).

For time complexity, when the number of nodes n scales up (Figure 4(a)), HOT achieves up to $70\times$ faster speed and $50\times$ scale up in n compared with baselines. When the number of graphs K scales up (Figure 4(b)), HOT achieves up to $360\times$ faster speed and $4\times$ scale up in K compared with baselines. Such substantial outperformance can be attributed to the hierarchical nature of HOT(i.e., the block-diagonal structure of $\boldsymbol{\mathcal{S}}$). In contrast, other baselines with dense alignment tensor are out-of-memory (OOM) when $K \geq 5$.

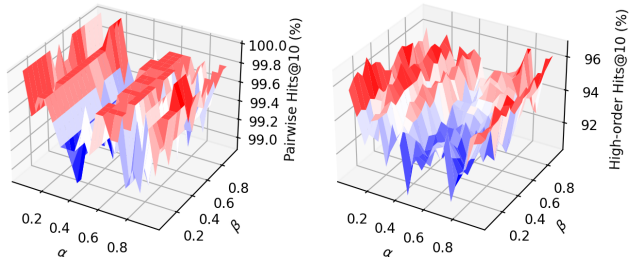
For space complexity, when the number of nodes n scales up (Figure 5(a)), HOT only consumes $\frac{1}{130}\times$ memory and scales up $10\times$ in n compared with baselines. When the number of graphs K scales up (Figure 5(b)), HOT only consumes $\frac{1}{1000}\times$ memory and scales up $5\times$ in K compared with baselines. Similarly, this is due to the utilization of block-diagonal property of $\boldsymbol{\mathcal{S}}$ achieving an exponential reduction in time and space complexities.

Besides, comparing HOT with different cluster number M in both Figures 4 and 5. When the cluster size $\overline{n}$ is small (e.g., left-most points with the number of nodes = 100), larger M results in higher time and space complexities as the cluster-level calculation is the dominant factor under such condition. However, when more nodes or networks are involved, both complexities are mostly dominated by the node-level alignment, where a larger M is preferred for better scalability.



(a) (b)

Figure 6: Convergence of the proximal point method: (a) Difference between two successive OT couplings (i.e., $\|\boldsymbol{\mathcal{S}}^{(t+1)} - \boldsymbol{\mathcal{S}}^{(t)}\|_1$); (b) MFGW distances along optimization.



(a) (b)

Figure 7: Hyperparameter study on DBLP(A)-500: (a) Pairwise Hits@10; (b) High-order Hits@10.

Convergence Analysis

We empirically validate the convergence guarantee in Proposition 3. We evaluate the difference between two successive OT coupling tensors $\boldsymbol{\mathcal{S}}$, i.e., $\|\boldsymbol{\mathcal{S}}^{(t+1)} - \boldsymbol{\mathcal{S}}^{(t)}\|_1$, and the values of the MFGW distance along the proximal point optimization. As shown in Figure 6, the objective function (i.e., MFGW distance) is non-increasing and the solution (i.e., $\boldsymbol{\mathcal{S}}$) converges along the proximal point iteration.

Hyperparameter Study

We study the sensitivity to hyperparameters α and β on the DBLP(A)-500 dataset, and results are shown in Figure 7. In general, the performance of HOT is stable in the whole feasible hyperparameter space. Both metrics are quite robust to the restart probability β but tend to decrease as α approaches 1. This is because attributes in DBLP(A)-500 dataset are quite informative, and the alignment almost exclusively depends on the structure when $\alpha \to 1$. Note that even under such an extreme case, HOT still achieves good performance.

RELATED WORK

Network alignment. Extensive efforts have been made for pairwise network alignment. Consistency-based methods (Koutra, Tong, and Lubensky 2013; Singh, Xu, and Berger 2008; Zhang and Tong 2016) are built upon the alignment consistency principle, which assumes similar node pairs to have similar alignment results. Embedding-based methods (Chu et al. 2019; Du, Yan, and Zha 2019; Zhang et al. 2021, 2020; Liu et al. 2016; Yan et al. 2021, 2023; Liu et al. 2022) generate low-dimensional node embeddings by pushing anchor node pairs as close as possible in the embedding space, enforced by contrastive loss (Jing, Park, and Tong 2021; Jing et al. 2022a). OT-based methods (Maretic et al. 2022; Xu, Luo, and Carin 2019; Xu et al. 2019; Zeng et al. 2023a; Zhao et al. 2020a,b; Wang et al. 2023) represent graphs as distributions and optimize the distribution alignment by minimizing the Wasserstein-like discrepancy. To reduce the time complexity, a line of work leverages the hierarchy of graphs by finding clusters in graphs (Zhang et al. 2019; Jing et al. 2022b; Wang, Dou, and Zhang 2022) to accelerate the pairwise alignment following a *coarsen-align-interpolate* strategy. For multi-network alignment, the transitivity constraint was first proposed to ensure the consistency between alignments for different network pairs (Chu et al. 2019; Zhang and Philip 2015; Zhou et al. 2020), but is hard to handle due to the non-convexity. The product graph, whose size becomes intractable in the multi-network setting, is adopted to model high-order alignment consistency (Du, Liu, and Tong 2021; Li et al. 2021). The low-rank approximation is explored for fast implicit solutions (Du, Liu, and Tong 2021; Li et al. 2021; Liu and Yang 2016), but high complexities are still inevitable when reconstructing the explicit solution.

Optimal transport on graphs. OT has achieved great success in handling geometric data such as graphs. Apart from network alignment reviewed above, OT has been applied in other graph-related tasks, including graph comparison (Titouan et al. 2019), graph clustering (Xu, Luo, and Carin 2019), graph compression (Garg and Jaakkola 2019) and graph representation learning (Vincent-Cuaz et al. 2021; Zeng et al. 2023b). The superiority of OT on graphs is rooted in its ability to capture the intrinsic geometry (Peyré, Cuturi, and Solomon 2016) and compare objects in different metric spaces, e.g., graphs (Mémoli 2011). While most OT literature is restricted to the two marginal cases, a few works (Beier, Beinert, and Steidl 2022; Fan et al. 2022; Pass 2015) study the multi-marginal setting suffering from the complex problem formulation and intractable computation. In this paper, we fill this gap by generalizing the FGW distance to the multi-marginal setting, followed by a fast proximal point solution with convergence to a local optimum.

CONCLUSION

In this paper, we study the multi-network alignment problem from the view of hierarchical multi-marginal optimal transport (MOT). To handle the high computational complexity, we explore the hierarchical structure of graph data and decompose the original problem into smaller cluster-level and node-level alignment subproblems. To depict high-order node relationships, a position-aware cost is first generated based on the unified random walk with restart (RWR) embeddings, and the multi-marginal fused Gromov-Wasserstein (MFGW) distance is further leveraged to align multiple networks jointly. A fast algorithm is proposed with guaranteed convergence to a local optimum, reducing both time and space complexities by an exponential factor in terms of the number of networks compared with the straightforward solution. Extensive experiments validate the effectiveness and scalability of the proposed HOT on the multi-network alignment task.

Acknowledgements

The ZZ and HT are partially supported by NSF (1947135, 2134079, 1939725, 2316233, and 2324770), DARPA (HR001121C0165), DHS (17STQAC00001-07-00), NIFA (2020-67021-32799) and ARO (W911NF2110088).

References

Alaux, J.; Grave, E.; Cuturi, M.; and Joulin, A. 2019. Unsupervised Hyper-alignment for Multilingual Word Embeddings. In *International Conference on Learning Representations*.

Beier, F.; Beinert, R.; and Steidl, G. 2022. Multi-Marginal Gromov-Wasserstein Transport and Barycenters. *arXiv preprint arXiv:2205.06725*.

Chen, L.; Gan, Z.; Cheng, Y.; Li, L.; Carin, L.; and Liu, J. 2020. Graph optimal transport for cross-domain alignment. In *International Conference on Machine Learning*, 1542–1553. PMLR.

Chu, X.; Fan, X.; Yao, D.; Zhu, Z.; Huang, J.; and Bi, J. 2019. Cross-network embedding for multi-network alignment. In *The world wide web conference*, 273–284.

Cuturi, M. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

Cuturi, M.; and Doucet, A. 2014. Fast computation of Wasserstein barycenters. In *International Conference on Machine Learning*, 685–693. PMLR.

Du, B.; Liu, L.; and Tong, H. 2021. Sylvester Tensor Equation for Multi-Way Association. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 311–321.

Du, B.; Zhang, S.; Yan, Y.; and Tong, H. 2021. New frontiers of multi-network mining: Recent developments and future trend. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 4038–4039.

Du, X.; Yan, J.; and Zha, H. 2019. Joint Link Prediction and Network Alignment via Cross-graph Embedding. In *IJCAI*, 2251–2257.

Fan, J.; Haasler, I.; Karlsson, J.; and Chen, Y. 2022. On the complexity of the optimal transport problem with graph-structured cost. In *International conference on artificial intelligence and statistics*, 9147–9165. PMLR.

Ferradans, S.; Papadakis, N.; Peyré, G.; and Aujol, J.-F. 2014. Regularized discrete optimal transport. *SIAM Journal on Imaging Sciences*, 7(3): 1853–1882.

Gao, J.; Huang, X.; and Li, J. 2021. Unsupervised graph alignment with wasserstein distance discriminator. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 426–435.

Garg, V.; and Jaakkola, T. 2019. Solving graph compression via optimal transport. *Advances in Neural Information Processing Systems*, 32.

Heimann, M.; Shen, H.; Safavi, T.; and Koutra, D. 2018. Regal: Representation learning-based graph alignment. In *Proceedings of the 27th ACM international conference on information and knowledge management*, 117–126.

Jing, B.; Feng, S.; Xiang, Y.; Chen, X.; Chen, Y.; and Tong, H. 2022a. X-GOAL: multiplex heterogeneous graph prototypical contrastive learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 894–904.

Jing, B.; Park, C.; and Tong, H. 2021. Hdmi: High-order deep multiplex infomax. In *Proceedings of the Web Conference 2021*, 2414–2424.

Jing, B.; Yan, Y.; Ding, K.; Park, C.; Zhu, Y.; Liu, H.; and Tong, H. 2023. STERLING: Synergistic Representation Learning on Bipartite Graphs. *arXiv preprint arXiv:2302.05428*.

Jing, B.; Yan, Y.; Zhu, Y.; and Tong, H. 2022b. Coin: Co-cluster infomax for bipartite graphs. *arXiv preprint arXiv:2206.00006*.

Koutra, D.; Tong, H.; and Lubensky, D. 2013. Big-align: Fast bipartite graph alignment. In *2013 IEEE 13th international conference on data mining*, 389–398. IEEE.

Li, C.; Wang, S.; Wang, Y.; Yu, P.; Liang, Y.; Liu, Y.; and Li, Z. 2019. Adversarial learning for weakly-supervised social network alignment. In *Proceedings of the AAAI conference on artificial intelligence*, 996–1003.

Li, Z.; Petegrosso, R.; Smith, S.; Sterling, D.; Karypis, G.; and Kuang, R. 2021. Scalable label propagation for multi-relational learning on the tensor product of graphs. *IEEE Transactions on Knowledge and Data Engineering*.

Liu, H.; and Yang, Y. 2016. Cross-graph learning of multi-relational associations. In *International Conference on Machine Learning*, 2235–2243. PMLR.

Liu, L.; Cheung, W. K.; Li, X.; and Liao, L. 2016. Aligning Users across Social Networks Using Network Embedding. In *IJCAI*, 1774–1780.

Liu, L.; Du, B.; Fung, Y. R.; Ji, H.; Xu, J.; and Tong, H. 2021. Kompare: A knowledge graph comparative reasoning system. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 3308–3318.

Liu, L.; Du, B.; Tong, H.; et al. 2019. G-finder: Approximate attributed subgraph matching. In *2019 IEEE international conference on big data (big data)*, 513–522. IEEE.

Liu, L.; Du, B.; Xu, J.; Xia, Y.; and Tong, H. 2022. Joint knowledge graph completion and question answering. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1098–1108.

Maretic, H. P.; El Gheche, M.; Chierchia, G.; and Frossard, P. 2019. GOT: an optimal transport framework for graph comparison. *Advances in Neural Information Processing Systems*, 32.

Maretic, H. P.; El Gheche, M.; Chierchia, G.; and Frossard, P. 2022. FGOT: Graph distances based on filters and optimal transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 7710–7718.

Maretic, H. P.; Gheche, M. E.; Minder, M.; Chierchia, G.; and Frossard, P. 2020. Wasserstein-based graph alignment. *arXiv preprint arXiv:2003.06048*.

Mémoli, F. 2011. Gromov–Wasserstein distances and the metric approach to object matching. *Foundations of computational mathematics*, 11: 417–487.

Mena, G.; Belanger, D.; Linderman, S.; and Snoek, J. 2018. Learning latent permutations with gumbel-sinkhorn networks. *arXiv preprint arXiv:1802.08665*.

Park, J.-D.; Tran, C.; Shin, W.-Y.; and Cao, X. 2022. GradAlign: Gradual Network Alignment via Graph Neural Networks (Student Abstract). In *Proceedings of the AAAI conference on artificial intelligence*, 13027–13028.

Pass, B. 2015. Multi-marginal optimal transport: theory and applications. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(6): 1771–1790.

Peyré, G.; Cuturi, M.; and Solomon, J. 2016. Gromov-wasserstein averaging of kernel and distance matrices. In *International Conference on Machine Learning*, 2664–2672.

Razaviyayn, M.; Hong, M.; and Luo, Z.-Q. 2013. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM Journal on Optimization*, 23(2): 1126–1153.

Scetbon, M.; Cuturi, M.; and Peyré, G. 2021. Low-rank sinkhorn factorization. In *International Conference on Machine Learning*, 9344–9354. PMLR.

Singh, R.; Xu, J.; and Berger, B. 2008. Global alignment of multiple protein interaction networks with application to functional orthology detection. *Proceedings of the National Academy of Sciences*, 105(35): 12763–12768.

Titouan, V.; Courty, N.; Tavenard, R.; and Flamary, R. 2019. Optimal transport for structured data with application on graphs. In *International Conference on Machine Learning*, 6275–6284. PMLR.

Tong, H.; Faloutsos, C.; and Pan, J.-Y. 2006. Fast random walk with restart and its applications. In *Sixth international conference on data mining (ICDM'06)*, 613–622. IEEE.

Vincent-Cuaz, C.; Vayer, T.; Flamary, R.; Corneli, M.; and Courty, N. 2021. Online graph dictionary learning. In *International Conference on Machine Learning*, 10564–10574.

Wang, Z.; Dou, J.; and Zhang, Y. 2022. Unsupervised sentence textual similarity with compositional phrase semantics. *arXiv preprint arXiv:2210.02284*.

Wang, Z.; Fei, W.; Yin, H.; Song, Y.; Wong, G. Y.; and See, S. 2023. Wasserstein-Fisher-Rao Embedding: Logical Query Embeddings with Local Comparison and Global Transport. *arXiv preprint arXiv:2305.04034*.

Xu, H.; Luo, D.; and Carin, L. 2019. Scalable gromov-wasserstein learning for graph partitioning and matching. *Advances in neural information processing systems*, 32.

Xu, H.; Luo, D.; Zha, H.; and Duke, L. C. 2019. Gromov-wasserstein learning for graph matching and node embedding. In *International Conference on Machine Learning*, 6932–6941.

Yan, Y.; Hu, Y.; Zhou, Q. Z.; Liu, L.; Zeng, Z.; Yuzhong, C.; Pan, M.; Chen, H.; Das, M.; and Tong, H. 2024. PACER: Network Embedding From Positional to Structural. In *The Web Conference 2024*.

Yan, Y.; Jing, B.; Liu, L.; Wang, R.; Li, J.; Abdelzaher, T.; and Tong, H. 2023. Reconciling Competing Sampling Strategies of Network Embedding. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yan, Y.; Liu, L.; Ban, Y.; Jing, B.; and Tong, H. 2021. Dynamic knowledge graph alignment. In *Proceedings of the AAAI conference on artificial intelligence*, 4564–4572.

Yan, Y.; Zhang, S.; and Tong, H. 2021. BRIGHT: A bridging algorithm for network alignment. In *Proceedings of the Web Conference 2021*, 3907–3917.

Yan, Y.; Zhou, Q.; Li, J.; Abdelzaher, T.; and Tong, H. 2022. Dissecting cross-layer dependency inference on multi-layered inter-dependent networks. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2341–2351.

Zeng, Z.; Zhang, S.; Xia, Y.; and Tong, H. 2023a. PARROT: Position-Aware Regularized Optimal Transport for Network Alignment. In *Proceedings of the ACM Web Conference 2023*, 372–382.

Zeng, Z.; Zhu, R.; Xia, Y.; Zeng, H.; and Tong, H. 2023b. Generative graph dictionary learning. In *International Conference on Machine Learning*, 40749–40769. PMLR.

Zhang, J.; and Philip, S. Y. 2015. Multiple anonymized social networks alignment. In *2015 IEEE International Conference on Data Mining*, 599–608. IEEE.

Zhang, S.; and Tong, H. 2016. Final: Fast attributed network alignment. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1345–1354.

Zhang, S.; Tong, H.; Jin, L.; Xia, Y.; and Guo, Y. 2021. Balancing Consistency and Disparity in Network Alignment. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2212–2222.

Zhang, S.; Tong, H.; Maciejewski, R.; and Eliassi-Rad, T. 2019. Multilevel network alignment. In *The World Wide Web Conference*, 2344–2354.

Zhang, S.; Tong, H.; Xia, Y.; Xiong, L.; and Xu, J. 2020. Nettrans: Neural cross-network transformation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 986–996.

Zhao, X.; Wang, Z.; Wu, H.; and Zhang, Y. 2020a. A relaxed matching procedure for unsupervised BLI. *arXiv preprint arXiv:2010.07095*.

Zhao, X.; Wang, Z.; Wu, H.; and Zhang, Y. 2020b. Semi-supervised bilingual lexicon induction with two-way interaction. *arXiv preprint arXiv:2010.07101*.

Zhou, Y.; Ren, J.; Jin, R.; Zhang, Z.; Dou, D.; and Yan, D. 2020. Unsupervised multiple network alignment with multinominal gan and variational inference. In *2020 IEEE International Conference on Big Data (Big Data)*, 868–877.