Learning Heat Diffusion for Network Alignment

Sisi Qu [1]   **Mengmeng Xu** [2]   **Bernard Ghanem** [2]   **Jesper Tegner** [1]

Abstract

Networks are abundant in the life sciences. Outstanding challenges include how to characterize similarities between networks, and in extension how to integrate information across networks. Yet, network alignment remains a core algorithmic problem. Here, we present a novel learning algorithm called evolutionary heat diffusion-based network alignment (EDNA) to address this challenge. EDNA uses the diffusion signal as a proxy for computing node similarities between networks. Comparing EDNA with state-of-the-art algorithms on a popular protein-protein interaction network dataset, using four different evaluation metrics, we achieve (i) the most accurate alignments, (ii) increased robustness against noise, and (iii) superior scaling capacity. The EDNA algorithm is versatile in that other available network alignments/embeddings can be used as an initial baseline alignment, and then EDNA works as a wrapper around them by running the evolutionary diffusion on top of them. In conclusion, EDNA outperforms state-of-the-art methods for network alignment, thus setting the stage for large-scale comparison and integration of networks.

1. Introduction

Considerable biological data related to signaling, metabolic, and regulatory relationships is usually represented as networks/graphs and is being generated at an increasing pace. However, such graphs tend to be challenging to process as they do not conform to grid-like structure or Euclidean geometry. A core algorithmic challenge in this domain is network alignment (NA), *i.e.* the process in which similarities between networks (leading to node correspondences) are quantitatively identified for the purpose of merging, information transfer. The task of NA has been extensively

---
[1]Living Systems Laboratory, King Abdullah University of Science and Technology (KAUST), Saudi Arabia. [2]Visual Computing Center, KAUST, Saudi Arabia. Correspondence to: Jesper Tegner <jesper.tegner@kaust.edu.sa>.
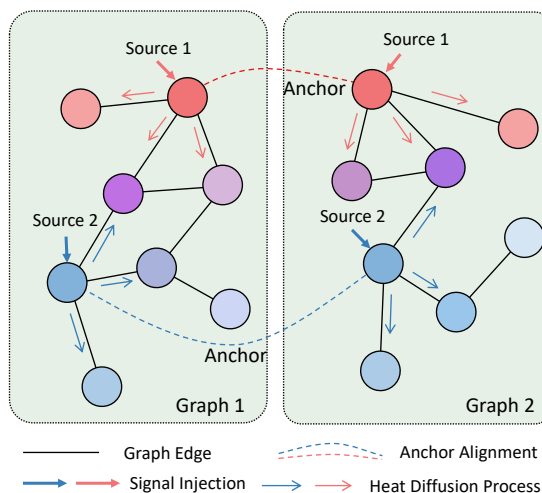
Figure 1. **Heat diffusion process from anchor node-pairs in two networks (graphs).** Specifically, in a PPI network, a node is a protein and an edge denotes an interaction. The signals of anchor nodes diffuse to all nodes, thus encoding both local structure and global structure of two networks.

studied over the last two decades. Protein-protein interaction (PPI) networks, where nodes are proteins that are connected with edges representing interactions, are of particular importance in biology. PPI networks contain valuable information for understanding protein functions, modular organization of cells, and system-level cellular processes (Hashemifar & Xu, 2014). Recently, the size of PPI networks has rapidly increased and numerous large-scale PPI maps have been produced by high throughput techniques in (Huttlin et al., 2015), (Huttlin et al., 2017), (Luck et al., 2019), *etc*. Comparative analysis of PPI networks has therefore become a key challenge as well as serving as a vehicle for data-driven integration of PPI networks.

Our work focuses on global network alignment (GNA) aiming at maximizing matches between entire networks first proposed by (Singh et al., 2008), with a specific application on PPI networks. GNA can be used to study interactomes and for understanding cross-species variations among PPI networks. Moreover, the GNA can facilitate the detection of functional orthologs. Here, we propose **evolutionary heat diffusion-based network alignment (EDNA)** to address the PPI network alignment problem. There are two stages in the EDNA pipeline. First, we obtain the baseline alignment
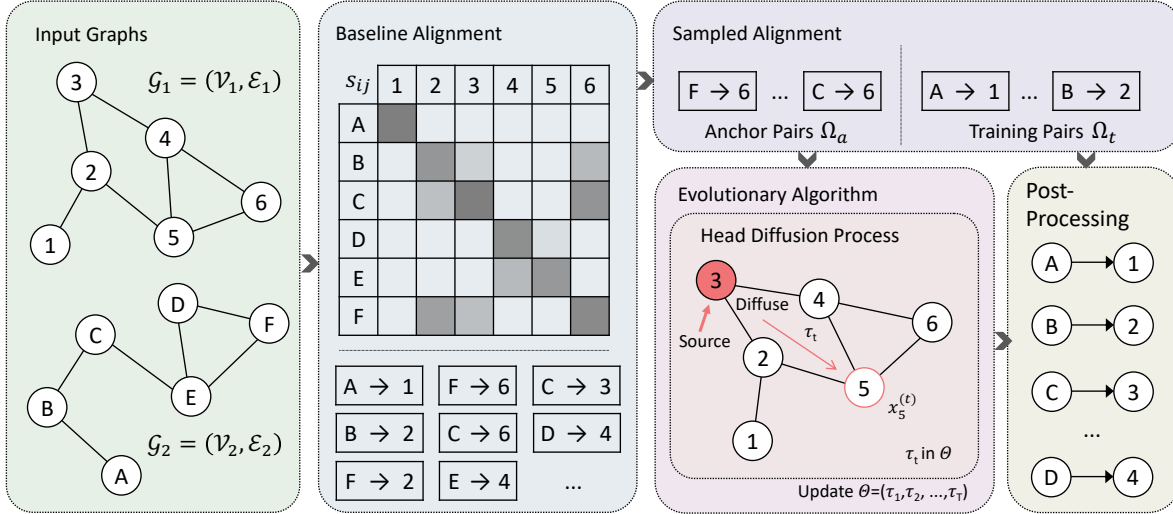
Figure 2. **Improved network alignment from evolutionary heat diffusion framework.** Using two input networks (graphs), we construct the baseline alignment based on embeddings or available alignments. Then we sample the alignment with highest precision as our anchor alignments. After applying heat diffusion using the signals of anchor alignment nodes to the whole graph T times, the evolutionary algorithm can be harnessed to fine-tune the diffusion parameter Θ to further improve performance.

calculated from extracted PPI network representations or other network alignment techniques, and on this basis we select nodes with highly confident alignments as anchor node-pairs. Secondly, evolutionary diffusion is deployed using those anchors with injected signals to diffuse information throughout the whole graph, and then a similarity matrix between the two networks is computed to enable network matching. Specifically, the heat diffusion process is applied on source anchor nodes as illustrated in Fig. 1. Importantly, EDNA is the first flexible framework that can be employed both on representations extracted from PPI networks by a variety of embedding techniques to obtain alignments, and also on any baseline alignment directly sampled from currently available network alignment algorithms to improve its performance (Fig. 2). We demonstrate the improved performance on a PPI network dataset using four different evaluation metrics with superior robustness, scalability, and computational efficiency.

2. Method

The EDNA framework is illustrated in Fig. 2. Let $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$ be two unweighted and undirected networks with node sets $\mathcal{V}_1$ and $\mathcal{V}_2$; edge sets $\mathcal{E}_1$ and $\mathcal{E}_2$, respectively. We construct the baseline alignment and sample the top γ confident alignment pairs to build the training set Ω_t and anchor set Ω_a (Sec. 2.1). We inject identical signals into paired nodes in Ω_a, and diffuse it for T times on $\mathcal{G}_1$ and $\mathcal{G}_2$, while the parameter Θ is optimized by evolutionary algorithms (Sec. 2.2). With diffused signal $\boldsymbol{x}_i^{(T)}$ on each node ν_i, we update the baseline alignment based on the signal similarity in post-processing (Sec. 2.3). Finally we analyse the computation complexity in Sec. 2.4.

2.1. Baseline Alignment Sampling

Many works have studied NA by exploring the feature representation from prior knowledge. Each node $\nu_i \in \mathcal{V}$ is represented by a embedding vector $\boldsymbol{e}_i$. One could also incorporate biological features into the PPI representations (*e.g.*, sequence similarity scores computed by BLAST) to enrich the encoded information. To align the networks, we take the node representation for two nodes $\nu_i \in \mathcal{V}_1, \nu_j \in \mathcal{V}_2$, apply an exponential kernel on their Euclidean distance in the embedding space, and take the output as the alignment confidence score, shown in Eq. 1.

$$s_{ij} = \exp(-\|\boldsymbol{e}_i - \boldsymbol{e}_j\|_2^2) \qquad (1)$$

By doing so, we have a list of node pairs (i, j) associated with confidence score s_{ij} as our baseline alignment. Worth-mentioning, EDNA can be flexibly integrated with other non-embedding NA methods. Alignments from those NA methods can be directly fed into our pipeline as input.

From the baseline alignment, we respectively sample α and β node pairs to form an anchor set Ω_a and a training set Ω_t. To include randomness in the alignment samples, we compute the mean μ_s and standard derivation σ_s of the all the confidence scores $s_{\cdot\cdot}$, and randomly sample $\gamma = \alpha + \beta$ confident (i, j)-pairs where $s_{i,j} \geq \mu_s + \sigma_s$.

2.2. Evolutionary Heat Diffusion Process

After injecting identical signals into those paired nodes in Ω_a and simulate T heat diffusion steps on both $\mathcal{G}_1$ and $\mathcal{G}_2$. The diffused value $\boldsymbol{x}_i^{(T)}$ depends on the neighbourhood of ν_i, reflecting the topology information. We leverage an evolutionary algorithm to learn the diffusion parameters.

We respectively initialize the anchor and non-anchor node values by identical signals and zeroes. In practice, we generate α unique vectors $\boldsymbol{f}_k \in \mathbb{R}^C, k = 1, 2, ..., \alpha$ and assign them to each of the paired nodes. We keep the signals on other nodes as zero-vectors. Eq. 2 and Eq. 3 show the details to assign the initial graph signals, where $\Omega_a(k)$ designates the k-th node-pair in Ω_a.

$$\forall \nu_i \in \mathcal{G}_1, \boldsymbol{x}_i^{(0)} = \begin{cases} \boldsymbol{f}_k, & \exists \Omega_a(k) = (i, j) \\ \boldsymbol{0}, & \text{otherwise} \end{cases} \quad (2)$$

$$\forall \nu_j \in \mathcal{G}_2, \boldsymbol{x}_j^{(0)} = \begin{cases} \boldsymbol{f}_k, & \exists \Omega_a(k) = (i, j) \\ \boldsymbol{0}, & \text{otherwise} \end{cases} \quad (3)$$

Thus, when we assign the identical signals to the 2α nodes, only the paired nodes have the same initial heat values.

Then, we simulate the heat diffusion process on the corresponding graphs. In the process, the neighbouring node information is aggregated. The diffusion process can be formulated under different assumptions. In this paper we use Eq. 4 (Thanou et al., 2017), where $\mathbf{A} \in \mathbb{R}^{N \times N}$ and $\mathbf{D} \in \mathbb{R}^{N \times N}$ are respectively the adjacency matrix and degree matrix of the graph, which has N nodes. $\mathbf{X}^{(t)} \in \mathbb{R}^{N \times C}$ is the stacking of all the node signal $\boldsymbol{x}_i^{(t)}$ on the graph. We use the element-wise activation function $\sigma(x) = [1 + \exp(-x)]^{-1}$ to create non-linearity and map the value to the range $(0, 1)$. τ_t is the diffusion duration in t-th iteration.

$$\mathbf{X}^{(t+1)} = \mathbf{X}^{(t)} + \sigma(\tau_t(\mathbf{D} - \mathbf{A})\mathbf{X}^{(t)}) \quad (4)$$

We leverage a genetic algorithm to learn the diffusion duration $\Theta = [\tau_1, ..., \tau_T]$. We take as the fitness function the node accuracy on train alignment, Ω_t. The node accuracy $f(\Theta)$ is the portion of positive alignments (e.g. $l == j$) over all training alignment pairs. Specifically, for each node pair (i, j) in Ω_t, the positive case is when the closest node ν_l in $\mathcal{G}_2$ of ν_i in $\mathcal{G}_1$ is exactly ν_j, shown in Eq. 5, where $\mathbf{1}\{\}$ is the indicator function.

$$f(\Theta) = \frac{1}{\beta} \sum_{(i,j) \in \Omega_t} \mathbf{1}\{j = \underset{l}{\operatorname{argmin}}(\|\boldsymbol{x}_i^{(T)} - \boldsymbol{x}_l^{(T)}\|_2^2)\} \quad (5)$$

After T times heat diffusion, we assume there is enough information transfer. Thus, the updated signal of each node encodes sufficient topology information and can be used to improve the alignment between the two networks.

2.3. Post-Processing

In post-processing, we reassign the baseline alignment based on the signal similarity in $\mathbf{X}^{(T)}$. For each node $\nu_i \in \mathcal{G}_1$, we keep the M most confident baseline alignments that connect ν_i to M candidate nodes in $\mathcal{G}_2$. Thus, we only compare $\boldsymbol{x}_i^{(T)}$ with M diffused signals and take the closest node as the updated alignment. By doing so, we can both increase the alignment accuracy and maintain a short alignment time.

2.4. Complexity Analysis

Baseline Alignment: The computation complexity of the baseline alignment varies depending on the network embedding method or other network alignment approaches used as preliminary results to feed into our framework.

Alignment Sampling: To sample the baseline alignment, we take $\gamma \leq N$ confident alignment pairs, then we randomly split them into the anchor alignment set and training alignment set. It takes $\mathcal{O}(N)$ times to sample the pairs.

Head Diffusion Process: Eq. 4 takes $N \times N \times C$ times operation for the matrix multiplication, causing $\mathcal{O}(CN^2)$ time complexity. However, this simulation can be accelerated by parallel computing on GPU, and the computation complexity can be reduced to $\mathcal{O}(\max\{C, N\})$.

Post-Processing: We compare M candidate node signals on $\mathcal{G}_2$ to N node signals on $\mathcal{G}_1$. Since the diffused signal is represented by C dimension vectors, this process takes $\mathcal{O}(MNC)$ complexity.

In conclusion, the total complexity of the heat-diffusion based graph alignment is $\mathcal{O}(TCN^2)$ based on a CPU implementation and $\mathcal{O}(MNC)$ based on a GPU implementation.

3. Experiments

Experimental Setup. All experiments are performed on an Intel(R) Xeon(R) CPU E5-2680v4 at 2.40GHz with 126GB RAM. We accelerate the heat diffusion process by CUDA 10.0 using PyTorch 1.1 (Python 3.7) on Tesla V100 (32GB).

Dataset. We use a PPI network commonly used in the graph alignment community, a subgraph of human PPI network with 3890 proteins as nodes, and 76,584 edges to represent protein-protein interactions (Breitkreutz et al., 2007).

Noise on graph. We synthesize a network $\mathcal{G}_2$ from a known network $\mathcal{G}_1 = \{\mathcal{V}_1, \mathcal{E}_1\}$. We permute the nodes and perturb the edges with noise level p_s. Assume $\mathcal{G}_1$ is associated with *adjacency matrix* $\mathbf{A}_1$. We take two steps to generate the *adjacency matrix* $\mathbf{A}_2$ of $\mathcal{G}_2$: we first multiply $\mathbf{A}_1$ by a random permutation matrix $\mathbf{P}$: $\mathbf{A}_1^{perm} = \mathbf{P}\mathbf{A}_1\mathbf{P}^\top$;we then remove edges in $\mathbf{A}_1^{perm}$ with probability p_s without disconnecting any nodes and finally get $\mathbf{A}_2$.

Evaluation Metrics. We adopt commonly used metrics for PPI NA: accuracy at top-1 and top-5 choices, edge correctness (EC), and symmetric substructure score (S^3).

Accuracy is calculated as the ratio of the number of correct alignments to the total number of groudtruth alignments. We consider both top-1 and top-5 accuracy (Acc@1 and Acc@5). *EC* is the percentage of edges in one network that are aligned to the edges in another (Kuchaiev et al., 2010). *EC* can be viewed as the measure of how much the edges

are conserved between the two graphs. S^3 is proposed in (Saraph & Milenković, 2014) with respect to both the source network and the target network. It penalizes the alignment mapping of denser regions to sparse regions and vise-versa.

Learnable heat diffusion on baseline alignments. To assess the performance of our method EDNA, we feed two different features, ndegree feature and struc2vec embedding, of PPI networks into our pipeline and calculate baseline alignment on them. struc2vec (Ribeiro et al., 2017) takes a hierarchy to measure node similarity at different scales for node embedding and preserves *structural* similarity of nodes. ndegree features are calculated using the degree information of every node's neighbors considering neighbors up to k hops from the original node (Heimann et al., 2018).

Also, the alignment from REGAL, a state-of-the-art NA method, is directly used as a baseline to show the capability of our method as a gerneralizable framework to boost performance of currently available alignments.

As shown in Tab. 1, EDNA drastically increases the performance (especially Acc@1) of baselines from different embedding methods. Taking Acc@1 as an example, EDNA reaches 92.6% from 65.0% for ndegree. Remarkably, our proposed method can significantly improve the performance of NA method REGAL from 59.6% to 92.6%.

Method	Acc@1	Acc@5	EC	S^3
Baseline				
Struc2vec	63.9	88.7	78.6	75.7
REGAL	59.6	77.4	49.3	46.3
ndegree	65.0	80.9	63.1	60.3
After EDNA				
Struc2vec	73.4	95.4	55.4	55.6
REGAL	92.6	96.5	97.7	98.5
ndegree	92.6	96.7	98.1	98.8

Table 1. **Accuracy of baseline alignments and EDNA for fixed** 5% **noise level.**

Then, we investigate the performance of baselines compared to that after deploying our EDNA framework in terms of accuracy under different noise levels from 1% up to 10%. Fig. 3 clearly shows that EDNA is consistently more accurate than baselines from different representations and NA methods with improved robustness, a testimony to its great potential for boosting NA performance.

Scalability with Dataset Size. On Erdös-Rényi networks with n= 500 to 8000 nodes and a constant average degree of 10, we test the scalability of EDNA. Notably, the diffusion process can be run on GPU's in parallel, significantly reducing the time cost. In Tab. 2, we show that the running time on CPU or GPU increases sub-quadratically with the size of the networks.

Ablation Study. We ablate the main components of EDNA on PPI networks with noise level 10%, including anchor-based diffusion and evolutionary algorithm. Tab. 3 reports
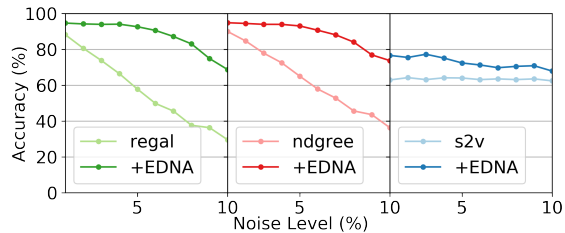

Figure 3. **Accuracy of baseline alignments and EDNA with varying noise levels from 1% to 10%**. The implementation with EDNA consistently outperforms baselines using different features from (struc2vec and ndegree) and NA method (REGAL).

Node Number	500	1000	2000	4000	8000
Top 1 Acc.	67.0	65.4	62.0	58.2	56.2
Top 5 Acc.	76.0	70.8	67.5	62.5	58.4
CPU (mean)	17.8	52.5	193	795	3041
CPU (std)	1.53	10.5	12.4	62.1	18.3
GPU (mean)	4.85	10.9	34.5	135	272
GPU (std)	0.46	0.17	0.59	3.7	1.4

Table 2. **Scalability of EDNA with network size.** The alignment accuracy (%) and running time (ms) is calculated for CPU or GPU respectively, using synthetic network with increasing number of nodes. With increasing network size the alignment accuracy decreases slightly, while the running time on CPU or on GPU increases. However, the simulation of the heat diffusion process on a GPU is much faster.

the results with each component disabled for five metrics. With only the diffusion block, Acc@1 can reach 52.75% from 35.63%, while using evolutionary diffusion, we can increase performance to 77.89%. Those two blocks are essential for maximizing alignment performance.

Method	Acc@1	Acc@5	EC	S^3
ndegree	35.63	57.09	27.77	23.53
Diffusion	52.75	73.54	56.96	56.26
EDNA	77.89	86.58	70.22	71.29

Table 3. **Ablation study of EDNA.** We use ndegree as a baseline alignment method to compare with the performance after applying diffusion and full EDNA. We report the results of four metrics respectively. p_s is set to 10%.

4. Conclusion

In this paper, we present a learnable evolutionary heat diffusion based algorithm for network alignment. On a general note, techniques for network alignment have a broad range of applications in computational biology. For example, in our hands, EDNA can be advantageous for integrating single cell RNA sequencing data, since single cells can be represented as nodes in a high-dimensional expression space, connected using k-nearest neighbor graphs.

References

Breitkreutz, B.-J., Stark, C., Reguly, T., Boucher, L., Breitkreutz, A., Livstone, M., Oughtred, R., Lackner, D. H., Bähler, J., Wood, V., et al. The biogrid interaction database: 2008 update. *Nucleic acids research*, 36 (suppl_1):D637–D640, 2007.

Hashemifar, S. and Xu, J. Hubalign: an accurate and efficient method for global alignment of protein–protein interaction networks. *Bioinformatics*, 30(17):i438–i444, 2014.

Heimann, M., Shen, H., Safavi, T., and Koutra, D. Regal: Representation learning-based graph alignment. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 117–126, 2018.

Huttlin, E. L., Ting, L., Bruckner, R. J., Gebreab, F., Gygi, M. P., Szpyt, J., Tam, S., Zarraga, G., Colby, G., Baltier, K., et al. The bioplex network: a systematic exploration of the human interactome. *Cell*, 162(2):425–440, 2015.

Huttlin, E. L., Bruckner, R. J., Paulo, J. A., Cannon, J. R., Ting, L., Baltier, K., Colby, G., Gebreab, F., Gygi, M. P., Parzen, H., et al. Architecture of the human interactome defines protein communities and disease networks. *Nature*, 545(7655):505–509, 2017.

Kuchaiev, O., Milenković, T., Memišević, V., Hayes, W., and Pržulj, N. Topological network alignment uncovers biological function and phylogeny. *Journal of the Royal Society Interface*, 7(50):1341–1354, 2010.

Luck, K., Kim, D. K., Lambourne, L., Spirohn, K., Begg, B. E., Bian, W., Brignall, R., Cafarelli, T., Campos-Laborie, F. J., Charloteaux, B., et al. A reference map of the human protein interactome. *bioRxiv*, pp. 605451, 2019.

Ribeiro, L. F., Saverese, P. H., and Figueiredo, D. R. struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 385–394, 2017.

Saraph, V. and Milenković, T. Magna: maximizing accuracy in global network alignment. *Bioinformatics*, 30(20): 2931–2940, 2014.

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

Thanou, D., Dong, X., Kressner, D., and Frossard, P. Learning heat diffusion graphs. *IEEE Transactions on Signal and Information Processing over Networks*, 3(3):484–499, 2017.