Title

Privacy Leakage In Graph Signal To Graph Matching Problems

Authors

Liu, Hang
Scaglione, Anna
Peisert, Sean

Publication Date

2024-04-19

DOI

10.1109/icassp48485.2024.10447364

Peer reviewed

PRIVACY LEAKAGE IN GRAPH SIGNAL TO GRAPH MATCHING PROBLEMS

Hang Liu[⋆]*, Anna Scaglione*[⋆] *and Sean Peisert*[†]

[*]Department of Electrical and Computer Engineering, Cornell Tech, Cornell University, NY USA
[†]Lawrence Berkeley National Laboratory, Berkeley, CA USA
Emails: hl2382@cornell.edu, as337@cornell.edu, sppeisert@lbl.gov

ABSTRACT

Graph matching over two known graphs is a method for de-anonymizing obscured node labels within an anonymous graph, finding the corresponding nodes in a second graph. In this paper, we consider a new case where a set of graph signals originate from a hidden graph. We want to match their components to a reference graph to reveal labels of asymmetric nodes. We refer to this as the graph-signal-to-graph matching (GS2GM) problem. We introduce a symmetry detection method to pinpoint the asymmetric nodes in the reference graph. Then, we adapt the existing blind graph matching algorithm, originally designed for asymmetric graphs, to align the detected nodes with signals generated from the target hidden graph. Furthermore, we establish sufficient conditions for perfect node de-anonymization through graph signals, showing that graph signals can leak substantial private information on the concealed labels of the underlying graph.

Index Terms— Graph matching, graph de-anonymization, network privacy, node identification, graph signal processing

1. INTRODUCTION

The emergence of expansive networks, e.g., in social media, infrastructure systems, and the Internet of Things, has led to an ever increasing influx of an unprecedented amount of data. While data publishers typically anonymize or randomize names and other identifying labels to safeguard the private information of local users, recent findings suggest that these conventional methods are still prone to privacy breaches. Adversaries can potentially discern a target user's identity from its local network connections by leveraging side information.

One notable attack method within this framework is called *seedless graph de-anonymization*. The aim is to infer labels within a wholly anonymized network by aligning the target nodes with a labeled reference graph sourced from public datasets, topology snapshots, etc. This de-anonymization strategy was first introduced in [1], where IMDB data served as the reference to identify the anonymized Netflix dataset. These authors further studied user de-anonymization in large social networks in [2]. Moreover, another pioneering work in [3] introduced an address anonymization framework for IP networks. The authors in [4] cast the de-anonymization challenge as a *graph matching* task that matches the nodes of the target and reference graphs by edge mismatch minimization. Maximum-a-posterior (MAP) estimators were introduced as tools tailored for social network de-anonymization, particularly with community structures [5, 6]. Sufficient conditions ensuring perfect node de-anonymization were derived for Erdös-Rényi (ER) random graphs in [4] and for stochastic block models in [5]. Moreover, [7] established a formula to measure the count of nodes that can possibly be de-anonymized via graph matching.

Current research relies on the topology of the anonymous graph for effective graph de-anonymization/matching, which can be resource-intensive or even unattainable in many real-world applications [8]. More commonly, attackers might directly observe interactions between nodes in an undisclosed graph, known as *graph signals*. Examples include opinion exchanges in social networks and nodal measurements in infrastructure systems and power grids. Recent studies have demonstrated that graph signals carry a plethora of information that can be leveraged for network analyses [9]. Particularly, [10] proposed a blind graph matching algorithm to match nodes from two unknown graphs using their associated graph signals, where the underlying graphs are assumed to have no symmetric structure.

Motivated by the above discussions, we study the privacy leakage in graph de-anonymization/matching when the problem is matching the graph signal components, that come from an undisclosed graph, to the nodes of a reference graph. We refer to this as the graph-signal-to-graph matching (GS2GM) problem. The resultant node identification can expose user identities by associating them with the labels of the reference [4]. We first show that symmetric structures pose barriers to node identification. To work around this problem, we introduce a low-complexity approximate symmetry detection method to distinguish symmetric nodes from the reference graph. Then, we adapt the graph matching algorithm in [10] to match the asymmetric nodes of the reference graph with the

This work was supported in part by the DoD-ARO under Grant No. W911NF2210228 and in part by the Director, Cybersecurity, Energy Security, and Emergency Response, Cybersecurity for Energy Delivery Systems program, of the U.S. Department of Energy, under contract DE-AC02-05CH11231.

graph signals. We conclude by analyzing the sufficient conditions for perfect de-anonymization of all asymmetric nodes and substantiate our analysis with experimental results.

2. SYSTEM MODEL

Let $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ be an undirected anonymous graph in which the labels of its nodes are concealed. Here, $\mathcal{V}_1$ and $\mathcal{E}_1$ denote the sets of nodes and edges, respectively. The number of nodes is $|\mathcal{V}_1| = N$, and we denote $\mathcal{V}_1 = [N] \triangleq \{1, 2, \cdots, N\}$. The adjacency matrix of $\mathcal{G}_1$ is denote by $\mathbf{A}^{(1)} \in \mathbb{R}^{N \times N}$, where $a^{(1)}_{kl} = a^{(1)}_{lk} > 0$ if and only if an edge (k, l) is present in $\mathcal{E}_1$. The Laplacian matrix of $\mathcal{G}_1$ is defined as $\mathbf{L}^{(1)} \triangleq \text{diag}(\mathbf{A}^{(1)}\mathbf{1}) - \mathbf{A}^{(1)}$, where $\mathbf{1}$ is the all-one vector.

Consider another N-node undirected and labeled graph $\mathcal{G}_2 = (\mathcal{V}_2 = [N], \mathcal{E}_2)$. When using its Laplacian matrix, denoted by $\mathbf{L}^{(2)}$, as a reference, the goal of the de-anonymization attack (a.k.a. node re-identification) is to determine a permutation function $\sigma(\cdot) : [N] \to [N]$. This function maps the node set $\mathcal{V}_2$ (or a subset of it) to $\mathcal{V}_1$ such that the permutation of $\mathcal{G}_2$ using $\sigma(\cdot)$ leads to a graph closely resembling $\mathcal{G}_1$ [4, 5]. Since the nodes of $\mathcal{G}_2$ possess known labels, an attacker can use $\sigma(\cdot)$ to identify the labels of the target nodes in $\mathcal{G}_1$. The node identification may then allow the attacker to infer private information on user identities [4].

Previous research on graph de-anonymization primarily relies on the adjacency or Laplacian matrix of Graph $\mathcal{G}_1$ to find an effective node matching. Such methods necessitate complete knowledge of the anonymous graph's topology. In contrast, we study a scenario where both the adjacency and Laplacian matrices of $\mathcal{G}_1$ remain unknown. In this context, an attacker seeks to de-anonymize the labels of the components of a collection of graph signals generated on $\mathcal{G}_1$.

Graph signal model. Consider the eigendecompositions of the two Laplacian matrices $\mathbf{L}^{(1)}$ and $\mathbf{L}^{(2)}$ as

$$\mathbf{L}^{(i)} = \mathbf{V}^{(i)}\mathbf{\Gamma}^{(i)}(\mathbf{V}^{(i)})^T, i = 1, 2, \tag{1}$$

where $\mathbf{\Gamma}^{(i)}$ is a diagonal matrix containing eigenvalues arranged in descending order $\gamma^{(i)}_1 \geq \gamma^{(i)}_2 \geq \cdots \geq \gamma^{(i)}_n = 0$, and $\mathbf{V}^{(i)} \in \mathbb{R}^{n \times n}$ is an orthogonal matrix containing the corresponding eigenvectors. As we assume no knowledge of $\mathbf{L}^{(1)}$, its eigendecomposition in (1) is subject to unknown permutations. In contrast, we observe a set of *filtered graph signals*, denoted by $\{\mathbf{y}_m\}^M_{m=1}$, which are generated over the nodes of $\mathcal{G}_1$ by an unknown graph filter. This graph filter can be expressed as a matrix polynomial of the Laplacian matrix $\mathbf{L}^{(1)}$:

$$\mathcal{H}(\mathbf{L}^{(1)}) = \sum_{t=0}^{T-1} h_t(\mathbf{L}^{(1)})^t = \mathbf{V}^{(1)} \left(\sum_{t=0}^{T-1} h_t(\mathbf{\Gamma}^{(1)})^t \right) (\mathbf{V}^{(1)})^T, \tag{2}$$

where T is the order of the graph filter, and $\{h_t\}$ are the filter coefficients. With (2), each observed signal vector $\mathbf{y}_m \in \mathbb{R}^{n \times 1}, \forall 1 \leq m \leq M$, is the output of $\mathcal{H}(\mathbf{L}^{(1)})$ excited by an input signal $\mathbf{x}_m \in \mathbb{R}^{n \times 1}$, as

$$\mathbf{y}_m = \mathcal{H}(\mathbf{L}^{(1)})\mathbf{x}_m + \mathbf{w}_m, \tag{3}$$

where $\mathbf{w}_m$ represents the unknown measurement noise of the m-th sample following the Gaussian distribution $\mathcal{N}(\mathbf{0}, \nu^2\mathbf{I}_n)$. We assume that $\mathbf{x}_m$ satisfies $\mathbb{E}[\mathbf{x}_m] = \mathbf{0}$ and $\mathbb{E}[\mathbf{x}_m(\mathbf{x}_m)^T] = \mathbf{I}_n, \forall m$. Substituting (1) into (2), the eigenvalues of $\mathcal{H}(\mathbf{L}^{(i)})$, often referred to as the frequency responses, are

$$\tilde{h}_k = \sum_{t=0}^{T-1} h_t(\gamma^{(1)}_k)^t, k \in [N]. \tag{4}$$

Note that the noiseless covariance matrix of $\mathbf{y}_m$ with $\mathbf{w}_m = \mathbf{0}$ is given by $\mathcal{H}(\mathbf{L}^{(1)})\mathcal{H}(\mathbf{L}^{(1)})^T$. Its eigendecomposition is given by

$$\mathcal{H}(\mathbf{L}^{(1)}) \left(\mathcal{H}(\mathbf{L}^{(1)}) \right)^T = \mathbf{V}^{(1)}\mathbf{\Lambda}^{(1)}(\mathbf{V}^{(1)})^T, \tag{5}$$

where $\mathbf{\Lambda}^{(1)} = \text{diag}([\lambda^{(1)}_1, \cdots, \lambda^{(1)}_N])$ contains the eigenvalues sequenced in descending order. It is worth noting that the eigenvalues $\{\lambda^{(1)}_n\}^N_{n=1}$ in (5) is a reshuffled arrangement of $\{\tilde{h}^2_k\}^N_{k=1}$ from (4), where the order is determined by the characteristic of the graph filter. For example, low-pass graph filters tend to focus their frequency responses on the lower graph frequencies, whereas high-pass graph filters amplify the higher graph frequencies. Interested readers can consult [9] and [10, Sect. III-A] for detailed examples of these filters. Here, we assume knowledge of the ordering of the frequency responses but not their actual values. In particular, the interchange between $\{\lambda^{(i)}_n\}^N_{n=1}$ and $\{\tilde{h}^2_k\}^N_{k=1}$ is specified by an index mapping ord$(\cdot)$ so that $\lambda^{(1)}_n = \tilde{h}^2_{\text{ord}(n)}, \forall n$. We note that the characteristics of the graph filter, such as whether it is low-pass or high-pass, can inform us about the order of the frequency responses. This assumption is considerably less stringent than that of knowing the exact values of the responses.

Graph symmetry and de-anonymizability. As shown in [11], graphs with symmetric structures, where nodes exhibit identical inner and outer structures, possess multiple graph automorphisms.[1] Hence, several equally optimal node permutations for de-anonymization exist [11]. Accordingly, two nodes $i, j \in [N]$ in $\mathcal{G}$ are symmetric (a.k.a. automorphically equivalent) if the node-swapping function σ with $\sigma(i) = j$, $\sigma(j) = i$, and $\sigma(k) = k, \forall k \neq i, j$, is an automorphism of $\mathcal{G}$. We use the notation $i \sim j$ to denote such symmetric nodes. For a graph $\mathcal{G}$ with N nodes, we denote the set of all symmetric nodes as $\mathcal{S}(\mathcal{G}) \triangleq \{i \in [N] : \exists j \in [N], j \neq i, i \sim j\}$. In contrast, the set of asymmetric nodes is denoted by $\mathcal{AS}(\mathcal{G}) \triangleq [N] \setminus \mathcal{S}(\mathcal{G})$, where $\setminus$ is the set difference operation.

Even when the graph topology of $\mathcal{G}_1$ is given, attackers, in the absence of further information, are unable to conclusively re-identify the true labels of the symmetric nodes in $\mathcal{S}(\mathcal{G}_1)$. In this work, we study the problem of de-anonymizing the asymmetric nodes of the target graph.

[1] An automorphism of a graph is a node permutation that yields an isomorphic (equivalent) graph [12].

Problem statement. We consider a general GS2GM problem where the attacker aims to infer the labels of graph signals corresponding to *all the asymmetric nodes* in $\mathcal{AS}(\mathcal{G}_1)$ based on the observed graph signals and the reference graph $\mathcal{G}_2$. Unless otherwise specified, $\mathcal{G}_2$ is assumed to be a graph isomorphism to $\mathcal{G}_1$, i.e., $\mathcal{G}_1$ and $\mathcal{G}_2$ are identical under an unspecified node permutation, denoted by $\sigma^\star(\cdot)$. Consequently, the challenge of computing $\sigma^\star$ is equivalent to matching the nodes of $\mathcal{G}_2$ and of the hidden $\mathcal{G}_1$. To formally state the objective, we define perfect de-anonymization as follows.

Definition 1 (Asymptotic perfect de-anonymization)**.** Given an unknown graph $\mathcal{G}_1$ and its isomorphic reference $\mathcal{G}_2$, consider a matching algorithm that uses M-sample graph signals $\{\mathbf{y}_m\}_{m=1}^M$ to produce a node permutation $\widehat{\sigma}(\cdot) : \mathcal{V}_2 \to \mathcal{V}_1$. This algorithm is said to achieve asymptotic perfect de-anonymization, provided that

$$\lim_{M\to\infty} \Pr(\widehat{\sigma}(n) = \sigma^\star(n), \forall n \in \mathcal{AS}(\mathcal{G}_2)) = 1. \tag{6}$$

Note that (6) requires the accurate alignment of the asymmetric nodes only, as de-anonymization of the symmetric nodes suffers from inevitable ambiguities in automorphisms. Our work focuses on addressing the following challenges:

1. Finding a computationally efficient strategy to identify $\mathcal{AS}(\mathcal{G}_2)$ for any given graph $\mathcal{G}_2$;
2. Developing an efficient de-anonymization algorithm to determine the node permutation $\widehat{\sigma}(\cdot) : \mathcal{V}_2 \to \mathcal{V}_1$ using the observed graph signals $\{\mathbf{y}_m\}$ and $\mathbf{L}^{(2)}$;
3. Analyzing the conditions to achieve asymptotic perfect de-anonymization.

As a final remark, we highlight that most of the existing work bases the de-anonymization analysis on specific probabilistic graph models, such as ER random graphs [1, 4] and stochastic block models [5, 6, 13]. Recognizing that such models might not adequately capture the complexities inherent in real-world networks, our approach is applicable to de-anonymizing an arbitrary graph $\mathcal{G}_1$ without the need of assuming its statistical model.

3. GRAPH SIGNAL TO GRAPH MATCHING

3.1. Proposed Method

We present a graph de-anonymization algorithm adapted from the method in [10] to compute the node permutation $\widehat{\sigma}(\cdot)$ using $\{\mathbf{y}_m\}_{m=1}^M$ and $\mathbf{L}^{(2)}$. For ease of notation, we represent the node permutation interchangeably by $\sigma(\cdot)$ and its equivalent permutation matrix $\mathbf{P} \in \{0,1\}^{N\times N}$, where $p_{kl} = 1$ if $\sigma(k) = l$ and $p_{kl} = 0$ otherwise. It is noteworthy that the graph matching algorithm from [10] was initially designed under the assumption that the two graphs do not have symmetric nodes. To align with our requirements, we incorporate a symmetry detection mechanism as follows.

Approximate Symmetry detection: We first estimate the symmetric nodes $\mathcal{S}(\mathcal{G}_2)$. Given that a graph with n nodes can have as many as $n!$ node permutations, the complexity of the exhaustive search for symmetric nodes, known as the graph automorphism problem, becomes prohibitively large complexity for large n. Here, we consider a polynomial-time method to find a *subset* of $\mathcal{S}(\mathcal{G}_2)$. For any i, j, let $\mathbf{P}^{(i,j)}$ be the swapping matrix that swaps the i-th and j-th columns of the identity matrix $\mathbf{I}_N$. For the graph $\mathcal{G}_2$, $i \sim j$ if $\mathbf{L}^{(2)} = (\mathbf{P}^{(i,j)})^T\mathbf{L}^{(2)}\mathbf{P}^{(i,j)}$. We determine symmetry by this condition. If $\mathbf{L}^{(2)} = (\mathbf{P}^{(i,j)})^T\mathbf{L}^{(2)}\mathbf{P}^{(i,j)}$, nodes i and j are added to the estimated symmetric node set $\widetilde{\mathcal{S}}(\mathcal{G}_2)$. This requires to evaluate a total of $N(N+1)/2$ pairs. Subsequently, the asymmetric node set is determined by $\widetilde{\mathcal{AS}}(\mathcal{G}_2) = [N] \setminus \widetilde{\mathcal{S}}(\mathcal{G}_2)$. As this method only identifies single-swap symmetric nodes, $\widetilde{\mathcal{AS}}(\mathcal{G}_2)$ is an overestimation (i.e., a superset) of $\mathcal{AS}(\mathcal{G}_2)$.

Eigenvector computation: We compute the eigenmatrices $\mathbf{V}^{(1)}$ and $\mathbf{V}^{(2)}$, which will be used for the subsequent node matching step. While $\mathbf{V}^{(2)}$ can be readily obtained from (1), we estimate the eigenmatrix $\mathbf{V}^{(1)}$ from the sample covariance of the graph signals in (3). Specifically, the sample covariance of $\{\mathbf{y}_m\}_m$ and its eigendecomposition are

$$\widehat{\mathbf{C}}_y = \frac{1}{M}\sum_{m=1}^{M} \mathbf{y}_m(\mathbf{y}_m)^T = \mathbf{U}^{(1)}\widehat{\mathbf{\Lambda}}^{(1)}(\mathbf{U}^{(1)})^T, \tag{7}$$

where $\mathbf{U}^{(1)}$ is orthogonal containing the sample eigenvectors, and $\widehat{\mathbf{\Lambda}}^{(1)} = \mathrm{diag}([\hat{\lambda}_1^{(1)}, \cdots, \hat{\lambda}_N^{(1)}])$ positions the sample eigenvalues in descending order. As M increases, we anticipate that $\mathbf{U}^{(1)}$ and $\{\hat{\lambda}_n^{(1)}\}$ provide an approximation close to $\mathbf{V}^{(1)}$ and $\{\lambda_n^{(1)}\}$ in (5), respectively. To ensure correspondence, we align the eigenvectors in $\mathbf{V}^{(2)}$ to match the order of their frequency responses with respect to $\mathcal{H}(\cdot)$. To this end, we shuffle the columns of $\mathbf{V}^{(2)}$ based on the indices delineated by the sequence $[\mathrm{ord}(1), \cdots, \mathrm{ord}(N)]$. The resulting reordered eigenmatrix is denoted by $\widetilde{\mathbf{V}}^{(2)}$.

Node matching: Following [10], we compute the node permutation matrix $\widehat{\mathbf{P}}$ by aligning the first K eigenvectors in $\mathbf{U}^{(1)}$ and $\widetilde{\mathbf{V}}^{(2)}$. The hyper-parameter K is determined by keeping the minimum spectral gap of the associated sample eigenvalues large [10, Sect. IV-D]. Denote by $\mathbf{U}_K^{(1)}$ and $\widetilde{\mathbf{V}}_K^{(2)}$ the submatrices consisting of the left K columns of $\mathbf{U}^{(1)}$ and $\widetilde{\mathbf{V}}^{(2)}$, respectively. To tackle the sign ambiguities inherent to eigendecompositions, we compute the permutation by taking the magnitudes of these eigenvectors: (cf. [10, Eq. (14)])

$$\widehat{\mathbf{P}} = \underset{\mathbf{P} \text{ is a permutation}}{\arg\max} \ \mathrm{tr}\left(\overline{\mathbf{U}}_K^{(1)}(\overline{\widetilde{\mathbf{V}}}_K^{(2)})^T\mathbf{P}\right), \tag{8}$$

where $\overline{\mathbf{U}}_K^{(1)}$ and $\overline{\widetilde{\mathbf{V}}}_K^{(2)}$ are the matrices with absolute values from $\mathbf{U}^{(1)}$ and $\widetilde{\mathbf{V}}^{(2)}$, respectively. The problem in (8) can be solved by off-the-shelf algorithms, such as the Hungarian method [14] or the greedy method in [10, Algorithm 2].

3.2. Condition for Asymptotic Perfect De-anonymization

We introduce a sufficient condition under which the proposed algorithm ensures asymptotic perfect de-anonymization de-

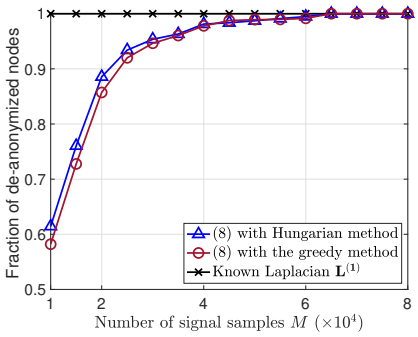

Fig. 1: De-anonymization accuracy for the ER graph.

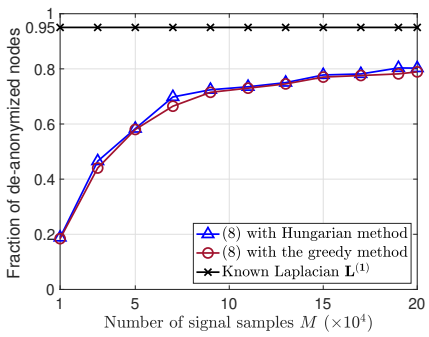

Fig. 2: De-anonymization over the *Facebook* network.

fined in Definition 1. Due to space constraints, the detailed proof is reserved for the extended version of this work.

Theorem 1. The node mapping $\widehat{\sigma}(\cdot)$ or equivalently $\widehat{\mathbf{P}}$ in (8) satisfies (6) if the following conditions are met:

1. The symmetry detection is accurate: $\widetilde{\mathcal{AS}}(\mathcal{G}_2) = \mathcal{AS}(\mathcal{G}_2)$;
2. $\{\mathbf{y}_m^{(i)}\}_{m=1}^M$ are independent, identically distributed (i.i.d.) and uniformly bounded almost surely.
3. Let $X_{i,j}$ be the (i,j)-th entry of $\overline{\mathbf{V}}_K^{(1)}(\widetilde{\overline{\mathbf{V}}}_K^{(2)})^T$, where $\overline{\mathbf{V}}_K^{(1)}$ is obtained from the error-free eigenvectors in (2) as $[\overline{\mathbf{V}}_K^{(1)}]_{kl} = |v_{kl}^{(1)}|$. It should be satisfied that
$$\rho \triangleq \min_{n\in\mathcal{AS}(\mathcal{G}_2)} \left(X_{n,\sigma^\star(n)} - \max_{\ell\neq\sigma^\star(n)} X_{n,\ell} \right) > 0. \quad (9)$$
4. The signal noise variance $\nu^2 < \frac{\sqrt{2}\rho}{8\sqrt{\sum_{k=1}^K 1/\delta_k^2}}$, where $\delta_k \triangleq \min\{\lambda_k^{(1)} - \lambda_{k+1}^{(1)}, \lambda_{k-1}^{(1)} - \lambda_k^{(1)}\}$ is the k-th spectral gap of the covariance matrix in (5).

Condition 2 can typically be met with i.i.d. and sub-Gaussian excitations. Condition 3 underscores the scenario where, in an error-free environment with $\mathbf{L}^{(1)}$ known, the true permutation $\sigma^\star(\cdot)$ maximizes the objective in (8). Motivated by this, we adopt ρ in (9) as a measure for assessing the graph de-anonymizability in the noiseless environment. The study in [15] reports that (9) asymptotically holds with large Gaussian models or large ER graphs. For a general case involving an arbitrary and unknown graph $\mathcal{G}_1$, ρ can be approximated using the sample eigenvectors in (7). Finally, Condition 4 guarantees accurate eigenvector estimation using the sample covariance given an adequate number of signal samples. This condition highlights the sensitivity of eigenvector estimation to signal noise, which is determined by the spectral gaps of the covariance matrix.

4. EXPERIMENTAL RESULTS

We evaluate the graph de-anonymization performance using two kinds of graphs: 1) the ER graph model and 2) the social network graph constructed from the *Facebook* dataset [16].

We begin by de-anonymizing a graph $\mathcal{G}_1$ generated by the ER model, consisting of $N = 50$ nodes with an edge probability of 0.4. The reference graph $\mathcal{G}_2$ is created by randomly shuffling the nodes of $\mathcal{G}_1$. We employ the opinion-dynamic model in [9] to design a low-pass graph filter as $\mathcal{H}(\mathbf{L}^{(1)}) = (\mathbf{I}_N + 0.1\mathbf{L}^{(1)})^{-1}$. We generate $\{\mathbf{y}_m\}$ by (3) with $\mathbf{x}_m$ drawn from $\mathcal{N}(\mathbf{0}, \mathbf{I}_N)$ and the noise variance set to 0.01. Fig. 1 plots the fraction of correctly de-anonymized nodes, with values ranging from $[0, 1]$. Here, the proposed method in (8) is solved by the Hungarian method [14] and the greedy method in [10, Algorithm 2] with K set to 20. The ideal approach with a known Laplacian matrix $\mathbf{L}^{(1)}$ is included for comparison. As the number of signal samples M rises, the de-anonymization accuracy of our algorithm converges to one. This trend corroborates our analytical findings.

Next, we evaluate the de-anonymization performance on the *Facebook* network, representing friendships among anonymous 348 users with $2,866$ edges. Applying the symmetry detection technique in Section 3.1, we identify that 45 out of 348 nodes are symmetric. Fig. 2 plots the de-anonymization accuracy over the remaining 303 nodes. Note that the condition in (9) is not met as $\rho \approx -0.45$. Therefore, even for the ideal case with an error-free Laplacian matrix $\mathbf{L}^{(1)}$, achieving asymptotic perfect de-anonymization is infeasible in this experiment. It shows that our proposed method successfully de-anonymizes 80% of the asymmetric nodes even though this network exhibits symmetry.

5. CONCLUSION

This paper studied graph de-anonymization of matching asymmetric nodes from graph signals to a reference graph. We proposed a method to approximately detect symmetric nodes and then employed the blind graph matching algorithm to match asymmetric nodes. Furthermore, the theoretical analysis delineates sufficient conditions to ensure asymptotic perfect de-anonymization. Simulation results validate our analysis and demonstrate the efficiency of the proposed algorithm. While our current method hinges on i.i.d. excitation signals to achieve perfect de-anonymization, a compelling direction for future research would be to examine how correlated signals, that are present in many real-world applications, influence de-anonymization accuracy.

6. REFERENCES

[1] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *IEEE Symposium on Security and Privacy*, 2008, pp. 111–125.

[2] ——, "De-anonymizing social networks," in *IEEE Symposium on Security and Privacy*, 2009, pp. 173–187.

[3] R. Pang, M. Allman, V. Paxson, and J. Lee, "The devil and packet trace anonymization," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, p. 29–38, Jan. 2006.

[4] P. Pedarsani and M. Grossglauser, "On the privacy of anonymized networks," in *Proc. ACM SIGKDD*, 2011, p. 1235–1243.

[5] E. Onaran, S. Garg, and E. Erkip, "Optimal de-anonymization in random graphs with community structure," in *Proc. 50th Asilomar Conf. Signals Syst. Comput.*, 2016, pp. 709–713.

[6] J. Zhang, S. Qu, Q. Li, H. Kang, L. Fu, H. Zhang, X. Wang, and G. Chen, "On social network de-anonymization with communities: A maximum a posteriori perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 3, pp. 2859–2874, Mar. 2023.

[7] B. Miao, S. Wang, L. Fu, and X. Lin, "De-Anonymizability of social network: Through the lens of symmetry," in *n Proc. ACM Mobihoc*, Oct. 2020, p. 71–80.

[8] G. Mateos, S. Segarra, A. G. Marques, and A. Ribeiro, "Connecting the dots: Identifying network structure via graph signal processing," *IEEE Signal Process. Mag.*, vol. 36, no. 3, pp. 16–43, May 2019.

[9] R. Ramakrishna, H.-T. Wai, and A. Scaglione, "A user guide to low-pass graph signal processing and its applications: Tools and applications," *IEEE Signal Process. Mag.*, vol. 37, no. 6, pp. 74–85, Nov. 2020.

[10] H. Liu, A. Scaglione, and H.-T. Wai, "Blind graph matching using graph signals," *ArXiv preprint arXiv:2306.15747*, 2023. [Online]. Available: http://arxiv.org/abs/2306.15747

[11] Q. Van Tran and H.-S. Ahn, "Further analysis on structure and spectral properties of symmetric graphs," *arXiv preprint arXiv:2203.01408*, 2022. [Online]. Available: https://arxiv.org/abs/2203.01408

[12] P. J. Cameron and Q. Mary, "Automorphisms of graphs," *Topics in Algebraic Graph Theory*, vol. 102, pp. 137–155, 2004.

[13] L. Fu, J. Zhang, S. Qu, H. Kang, X. Wang, and G. Chen, "Measuring social network de-anonymizability by means of morphism property," *IEEE/ACM Trans. Netw.*, vol. 30, no. 6, pp. 2744–2759, Dec. 2022.

[14] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, Mar. 1957.

[15] Y. Wu, J. Xu, and S. H. Yu, "Settling the sharp reconstruction thresholds of random graph matching," *IEEE Trans. Inf. Theory*, vol. 68, no. 8, pp. 5391–5417, Aug. 2022.

[16] J. J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *Proc. Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 548–556.