Journal of Machine Learning Research 22 (2021) 1-54 Submitted 1/21; Revised 7/21; Published 10/21

Graph Matching with Partially-Correct Seeds

Liren Yu YU827@PURDUE.EDU
Elmore Family School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907, USA

Jiaming Xu JX77@DUKE.EDU
The Fuqua School of Business
Duke University
Durham, NC 27708, USA

Xiaojun Lin LINX@PURDUE.EDU
Elmore Family School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907, USA

Editor: Gabor Lugosi

Abstract

Graph matching aims to find the latent vertex correspondence between two edge-correlated graphs and has found numerous applications across different fields. In this paper, we study a seeded graph matching problem, which assumes that a set of seeds, i.e., pre-mapped vertex-pairs, is given in advance. While most previous work requires all seeds to be correct, we focus on the setting where the seeds are partially correct. Specifically, consider two correlated graphs whose edges are sampled independently from a parent Erdős-Rényi graph $\mathcal{G}(n,p)$. A mapping between the vertices of the two graphs is provided as seeds, of which an unknown β fraction is correct. We first analyze a simple algorithm that matches vertices based on the number of common seeds in the 1-hop neighborhoods, and then further propose a new algorithm that uses seeds in the 2-hop neighborhoods. We establish non-asymptotic performance guarantees of perfect matching for both 1-hop and 2-hop algorithms, showing that our new 2-hop algorithm requires substantially fewer correct seeds than the 1-hop algorithm when graphs are sparse. Moreover, by combining our new performance guarantees for the 1-hop and 2-hop algorithms, we attain the best-known results (in terms of the required fraction of correct seeds) across the entire range of graph sparsity and significantly improve the previous results in Kazemi et al. (2015); Lubars and Srikant (2018) when $p \geq n^{-5/6}$. For instance, when p is a constant or $p = n^{-3/4}$, we show that only $\Omega(\sqrt{n \log n})$ correct seeds suffice for perfect matching, while the previously best-known results demand $\Omega(n)$ and $\Omega(n^{3/4} \log n)$ correct seeds, respectively. Numerical experiments corroborate our theoretical findings, demonstrating the superiority of our 2-hop algorithm on a variety of synthetic and real graphs.

Keywords: graph matching, Erdős-Rényi graphs, partially-correct seeds, 2-hop witnesses, large deviation analysis

1. Introduction

Given a pair of two edge-correlated graphs, graph matching (also known as network alignment) aims to find a bijective mapping between the vertex sets of the two graphs so that their edge sets are maximally aligned. This is a ubiquitous but difficult problem arising in many important applications, such as social network de-anonymization (Narayanan and Shmatikov, 2009), computational biology (Singh et al., 2008; Kazemi et al., 2016), computer vision (Conte et al., 2004; Schellewald and Schnörr, 2005), and natural language processing (Haghighi et al., 2005). For instance, from one anonymized version of the "follow" relationships graph on the Twitter microblogging service, researchers were able to re-identify the users by matching the anonymized graph to a correlated cross-domain auxiliary graph, i.e., the "contact" relationships graph on the Flickr photo-sharing service, where user identities are known (Narayanan and Shmatikov, 2009).

Existing graph matching algorithms can be classified into two categories, seedless and seeded matching algorithms. Seedless matching algorithms only use the topological information and do not rely on any additional side information. Various seedless matching algorithms have been proposed based on either degree information (Dai et al., 2018; Ding et al., 2021), spectral method (Umeyama, 1988; Cour et al., 2007; Feizi et al., 2019; Fan et al., 2019a,b), random walk (Gori et al., 2005), convex relaxations (Aflalo et al., 2015; Fiori and Sapiro, 2015; Lyzinski et al., 2016; Dym et al., 2017; Bernard et al., 2018), or non-convex methods (Zaslavskiy et al., 2008; Fiori et al., 2013; Vogelstein et al., 2015; Yu et al., 2018; Maron and Lipman, 2018; Zhang et al., 2019; Xu et al., 2019). However, to the best of our knowledge, these algorithms either only provably succeed when the fraction of edges that differ between the two graphs is low, i.e., on the order of $O\left(1/\log^2 n\right)$ (Ding et al., 2021) or require at least quasi-polynomial runtime ($n^{O(\log n)}$) (Barak et al., 2018; Cullina and Kiyavash, 2016, 2017; Cullina et al., 2019), where n is the number of vertices in one graph. The only exception is the neighborhood tree matching algorithm recently proposed in Ganassali and Massoulié (2020), which can output a partially-correct matching in polynomial-time only when two graphs are sparse and differ by a constant fraction of edges.

The other category is seeded matching algorithms (Pedarsani and Grossglauser, 2011; Yartseva and Grossglauser, 2013; Korula and Lattanzi, 2014; Lyzinski et al., 2013; Fishkind et al., 2018; Shirani et al., 2017; Mossel and Xu, 2019; Chiasserini et al., 2016). These algorithms require "seeds", which are a set of pre-mapped vertex-pairs. Let G_1 and G_2 denote two graphs. For each pair of vertices (u, v) with u in G_1 and v in G_2, a seed (w, w') is called a *1-hop witness* for (u, v) if w is a neighbor of u in G_1 and w' is a neighbor of v in G_2. The basic idea of seeded matching algorithms is that a candidate pair of vertices are expected to have more witnesses if they are a true pair than if they are a fake pair. Assuming that the seeds are correct, seeded matching algorithms can find the correct matching for the remaining vertices more efficiently than seedless matching algorithm. In social network de-anonymization, such initially matched seeds are often available, thanks to users who have explicitly linked their accounts across different social networks. For other applications, the seeds can be obtained by prior knowledge or manual labeling.

However, most existing seeded matching algorithms crucially rely on all seeds being correct, which is often difficult to guarantee in practice. For example, the seeds may be

provided by seedless matching algorithms, which will likely produce some incorrect seeds. To overcome this limitation, Kazemi et al. (2015) and Lubars and Srikant (2018) extend the idea of seeded matching algorithms to allow for incorrect seeds. In particular, Kazemi et al. (2015) proposes a NoisySeeds algorithm, which uses percolation (Janson et al., 2012; Yartseva and Grossglauser, 2013) to grow the number of 1-hop witnesses from partially-correct seeds and iteratively matches pairs whose number of witnesses exceeds a threshold r. However, NoisySeeds is very sensitive to the choice of the threshold r and matching errors, and thus is only guaranteed to perform well when the graphs are very sparse. More specifically, when the two graphs are correlated Erdős-Rényi graphs, whose edges are independently sub-sampled with probability s from a *parent* Erdős-Rényi graph $\mathcal{G}(n,p)$, and when β fraction of seeds are correct, it is shown in Kazemi et al. (2015) that NoisySeeds with the best choice of threshold $r=2$ can correctly match all but $o(n)$ vertex-pairs with high probability, provided that $n^{-1} \ll p \leq n^{-\frac{5}{6}-\epsilon}$ for $\epsilon \in (0, 1/6)$, and

$$\beta \geq \frac{1}{2n^2p^2s^4}. \tag{1}$$

However, for denser graphs with $p \geq n^{-\frac{5}{6}}$, no performance guarantees are established in Kazemi et al. (2015) for the setting with incorrect seeds.

In contrast, Lubars and Srikant (2018) proposes a different algorithm that uses the numbers of 1-hop witnesses for each candidate pair of vertices as weights, and then uses Greedy Maximum Weight Matching (GMWM) to find the vertex correspondence between the two graphs such that the total number of witnesses is large. Lubars and Srikant (2018) shows that their 1-hop algorithm can work over a much wider range of p (up to $p \leq \frac{3}{8}$) than Kazemi et al. (2015), and it can correctly match all vertices with high probability if

$$\beta \geq \max\left\{\frac{16\log n}{nps^2},\ \frac{8}{3}p\right\}. \tag{2}$$

In order to illustrate the limitations of these existing results, we plot in Fig. 1 the scalings corresponding to the two conditions (1) and (2), as the black dotted curve and green dashed curve, respectively, where the x-axis is the graph sparsity p (which is bounded away from 1 and much greater than n^{-1}) and the sampling probability s is a constant. We observe that, when the graphs are sparse, condition (2) ($\beta = \Omega\left(\log n/np\right)$) requires substantially more correct seeds than condition (1) ($\beta = \Omega\left(1/n^2p^2\right)$), suggesting that the 1-hop algorithm in Lubars and Srikant (2018) is suboptimal. However, condition (1) only extends to $p \leq n^{-5/6}$, and is not applicable to denser graphs. When the graphs are dense, condition (2) requires β to increase proportionally in p. In particular, when p is a constant, condition (2) demands a constant fraction of correct seeds. Such a requirement seems rather stringent as well. In summary, the existing conditions on the required number of correct seeds are either pessimistic or only applicable to very sparse graphs. Since the number of correct seeds is often limited in practice, it is of paramount importance in both theory and practice to understand how to better utilize partially-correct seeds to attain more accurate matching results for both sparse and dense graphs.

In this paper, we propose a new algorithm based on the number of j-hop witnesses and establish performance guarantees for the 1-hop and 2-hop algorithms, significantly relaxing

the existing requirements in (1) and (2). Specifically, we first provide a much tighter analysis than Lubars and Srikant (2018), showing that the 1-hop algorithm can correctly match all vertices with high probability, provided that

$$\beta \geq \max\left\{\frac{45\log n}{np(1-p)^2s^2},\, 30\sqrt{\frac{\log n}{n(1-p)^2s^2}}\right\}. \tag{3}$$

Moreover, we show that the 2-hop algorithm can exactly match all vertices with high probability, provided that $np^2 \leq (\log n)^{-1}$, $nps^2 \geq 128\log n$, and

$$\beta \geq \max\left\{\frac{600\log n}{n^2p^2s^4},\, 600\sqrt{\frac{\log n}{ns^4}},\, 600\sqrt{\frac{np^3(1-s)\log n}{s}}\right\}. \tag{4}$$

See Section 4 for intuitive interpretations of the various terms in (4).

The new conditions (3) and (4), are also plotted in Fig. 1 as the solid red and blue curves, respectively, to illustrate the improvement compared to the previous conditions (1) and (2). (Note that there is a factor $1-s$ in our condition (4). As a consequence, the corresponding blue curve has two branches: the top one holds for $s < 1$ and the bottom one holds for $s = 1$.)

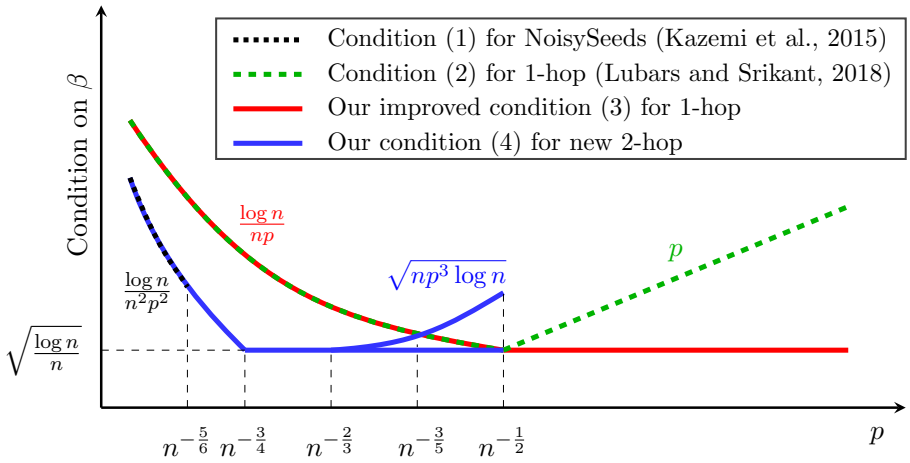


Figure 1: Comparison of the requirements on the fraction β of correct seeds, when s is a fixed constant and p is bounded away from 1. The lower the curve, the fewer correct seeds it requires.

From Fig. 1, we can see that by combining our two conditions (3) and (4) (i.e., the lower envelope of blue and red solid curves), we attain the lowest requirements on the number of correct seeds across the entire range of graph sparsity p, and significantly improve the previous conditions when $p \geq n^{-5/6}$. In particular,

- Comparing the green dashed curve with the red solid curve, we see that when $p \gg n^{-1/2}$ our condition (3) requires many fewer correct seeds than (2) for the 1-hop

algorithm to succeed. For instance, when p is a constant, only $\Omega(\sqrt{n \log n})$ correct seeds suffice for the 1-hop algorithm to achieve perfect matching according to our condition (3), while the previous condition (2) requires $\Omega(n)$ correct seeds.

- Comparing the green dashed curve with the blue solid curve, we see that when $p \ll n^{-1/2}$ for $s = 1$ or when $p \ll n^{-3/5}$ for $s < 1$, our condition (4) also requires substantially fewer correct seeds than (2). This shows that our new 2-hop algorithm is significantly better than the 1-hop algorithm when the graphs are sparse. For instance, when $p = n^{-3/4}$, only $\Omega(\sqrt{n \log n})$ correct seeds suffice for our 2-hop algorithm to achieve perfect matching, while the 1-hop algorithm requires $\Omega\left(n^{3/4} \log n\right)$ correct seeds.

- Comparing the black dotted curve with the blue solid curve, we see that our condition (4) is comparable to condition (1) when $p \ll n^{-5/6}$. However, our condition (4) continues to hold up to $p \ll n^{-1/2}$. This shows that our 2-hop algorithm enjoys competitive performance compared to NoisySeeds when graphs are very sparse, but is more versatile and continues to perform well over a much wider range of graph sparsity.

Furthermore, our results precisely characterize the graph sparsity at which the 2-hop algorithm starts to outperform 1-hop. This reveals an interesting and delicate trade-off between the *quantity* and the *quality* of witnesses: while the 2-hop algorithm exploits more seeds as witnesses than the 1-hop algorithm, the 2-hop witnesses can also be less discriminating (as they are further away from the node-pair under consideration). Thus, while the increased quantity helps when the graphs are sparse, the decreased quality can confuse the matching algorithm when the graphs are dense (e.g., even the fake pairs are likely to have many 2-hop witnesses).

Our results also significantly outperform the existing performance guarantees for polynomial-time seedless graph matching algorithms. The best known polynomial-time seedless algorithms require $1 - s = o(1)$ (Ding et al., 2021) to achieve perfect matching. The neighborhood tree matching algorithm proposed in Ganassali and Massoulié (2020) only provably outputs partially-correct matching when np and s are very close to 1. Compared to polynomial-time seedless algorithms, our proposed algorithm with enough seeds can tolerate a constant s much lower than 1.

Finally, using numerical experiments on both synthetic and real graphs, we show that our 2-hop algorithm significantly outperforms the state-of-the-art. Specifically, when the initial seeds are randomly chosen, the 2-hop algorithm significantly outperforms the 1-hop algorithm in Lubars and Srikant (2018) on sparse graphs, which agrees with our theoretical analysis. Further, the performance of our 2-hop algorithm is comparable to the NoisySeeds algorithm when the synthetic graphs are very sparse, and much better than the NoisySeeds algorithm on other graphs. When there are enough seeds, our experiments confirm that our 2-hop algorithm also outperforms the state-of-the-art polynomial-time seedless algorithms. Our 2-hop algorithm is also much more robust to power-law degree variations in real graphs than the NoisySeeds algorithm. Moreover, we conduct an experiment on matching 3D deformable shapes in which the initial seeded mapping is generated by a seedless algorithm (instead of randomly chosen). We demonstrate that our 2-hop algorithm drastically boosts

the matching accuracy by cleaning up most initial matching errors, and the performance enhancement is more substantial than the 1-hop algorithm and NoisySeeds algorithm. Computationally, our 2-hop algorithm is comparable to the 1-hop and NoisySeeds algorithm and runs efficiently on networks with $\sim 10K$ nodes on a single PC, and can potentially scale up to even larger networks using parallel implementation.

In passing, we remark that although we focus on matching two graphs of the same number of vertices, our 2-hop algorithm can be directly applied to matching two graphs of different sizes and return an accurate correspondence between nodes in the common subgraph of the two graphs. Indeed, the simulation results with real data in Section 6.2 show that our 2-hop algorithm still achieves outstanding matching performance, even when two graphs are of very different sizes.

1.1 Key Ideas and Analysis Techniques

Our improved performance guarantees for perfect matching exploit several key ideas and analysis techniques, which we present below and will elaborate further in later sections. For ease of discussion, we assume the true vertex correspondence between the two graphs is given by the identity permutation. We use $\pi : [n] \to [n]$ to denote the initial seed mapping. Then, each seed $(i, \pi(i))$ is correct if $\pi(i) = i$ and incorrect otherwise.

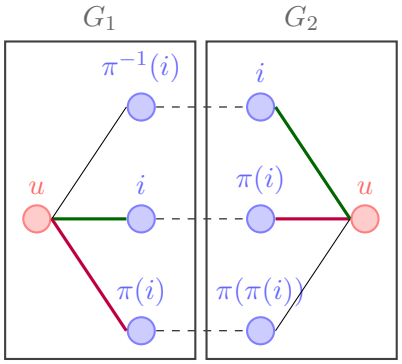


Figure 2: The two green/purple edges are correlated, because they correspond to the same edge in the parent graph. Thus, the event that an incorrect seed $(i, \pi(i))$ becomes a 1-hop witness for (u, u) is dependent on the events that $(\pi^{-1}(i), i)$ and $(\pi(i), \pi(\pi(i)))$ become 1-hop witnesses for (u, u).

To obtain a much tighter condition than (2) for the success of the 1-hop algorithm, our key observation is that, when counting the number of witnesses for the true pairs, the analysis of Lubars and Srikant (2018) only considers the correct seeds and ignores the incorrect seeds. It is non-trivial to consider the incorrect seeds for a true pair, because the events that the incorrect seeds become witnesses depend on each other. In particular, the event that an incorrect seed $(i, \pi(i))$ becomes a 1-hop witness for true pair (u, u) is dependent on the events that $(\pi^{-1}(i), i)$ and $(\pi(i), \pi(\pi(i)))$ become 1-hop witnesses for (u, u), as these events may involve the same edges in the parent graph (See Fig. 2 for an illustrating example). Our analysis takes into account the incorrect seeds for the true pairs

and carefully deals with the above dependency issue using concentration inequalities for dependent random variables (Janson, 2004). See Section 5.1 for details.

Further, to better utilize seeds in sparse graphs, our key idea is to match vertices by comparing the number of witnesses in the 2-hop neighborhoods. In sparse graphs, the number of 1-hop witnesses for most vertices will be very low. (For example, when $p = O(n^{-1/2})$, $\beta = O(n^{-1/2})$, and s is a constant, the number of 1-hop witnesses for even a true pair is about $\beta nps^2 + (1-\beta)np^2s^2 = O(1)$.) Therefore, it will be difficult to use 1-hop witnesses alone to distinguish true pairs from fake pairs. In contrast, the number of 2-hop witnesses will be much larger. Thus, compared to the algorithm in Lubars and Srikant (2018) that uses only 1-hop witnesses, our 2-hop algorithm can leverage more witnesses to distinguish the true pairs from the fake pairs. The idea of using multi-hop neighborhoods to match vertices is analyzed previously in Mossel and Xu (2019) when all seeds are correct. In comparison, our results on the 2-hop algorithm make several significant contributions. First, our analysis with incorrect seeds is considerably more challenging, as we need to take care of the dependency on the size of the 1-hop neighborhood and the dependency between incorrect seeds. Unfortunately, unlike the setting in the previous paragraph, here directly using the results of Janson (2004) will not work because the number of dependencies that we have to deal with may be very large (see Section 5.2 for details). Instead we deal with the dependency issues by first conditioning on the 1-hop neighborhood; and then analyzing different seeds according to different situations and applying the concentration inequalities for dependent random variables (Janson, 2004). Second, our condition (4) characterizes the influence of the incorrect seeds and reveals the delicate behavior of the 2-hop algorithm. In particular, we show that the 2-hop algorithm requires at least $\Omega(\sqrt{n \log n})$ correct seeds, irrespective of the graph sparsity. Also, somewhat surprisingly, we discover that when $s < 1$, the 2-hop algorithm may require more seeds as p increases from $n^{-2/3}$, due to the larger fluctuation of 1-hop neighborhood sizes. All these new phenomenons are absent when seeds are all correct and thus are not captured by the theoretical results in Mossel and Xu (2019). Third, the computational complexity of the algorithm in Mossel and Xu (2019) is $O(n^3)$, which is much higher than that of our algorithm – $O(n^\omega + n^2 \log n)$, where n^ω with $2 \le \omega \le 2.373$ denotes the time complexity for $n \times n$ matrix multiplication (see Section 3 for detail).

2. Model

In this section, we formally introduce the model and the graph matching problem with partially-correct seeds.

We use the $\mathcal{G}(n, p; s)$ graph model proposed by Pedarsani and Grossglauser (2011), which has been widely used in the study of graph matching. Let G_0 denote the parent graph with n vertices $\{1, 2, ..., n\} \triangleq [n]$. The parent graph G_0 is generated from the Erdős-Rényi model $\mathcal{G}(n, p)$, i.e., we start with an empty graph on n vertices and connect any pair of two vertices independently with probability p. Then, we obtain a subgraph G_1 by sampling each edge of G_0 into G_1 independently with probability s. Repeat the same sub-sampling process independently and relabel the vertices according to an *unknown* permutation $\pi^* : [n] \to [n]$ to construct another subgraph G_2. Throughout the paper, we denote a vertex-pair by (u, v),

where $u \in G_1$ and $v \in G_2$. For each vertex-pair (u, v), if $v = \pi^*(u)$, then (u, v) is a true pair; if $v \neq \pi^*(u)$, then (u, v) is a fake pair.

As a motivating example, the parent graph G_0 can be some underlying friendship network among n persons, while G_1 is the Flicker contact network and G_2 is a Twitter follow network among these n persons.

Prior literature proposes various algorithms to recover π^* based on G_1 and G_2. The output of these graph matching algorithms can be interpreted as a set of partially correct seeds. Taking these partially correct seeds as input, we wish to efficiently correct all of the errors. However, it is difficult to perfectly model the correlation between the output of these algorithms and the graphs. One way to get around this issue is to treat these partially correct seeds as adversarially chosen and to design an algorithm that with high probability corrects all errors for all possible initial error patterns. However, the existing theoretical guarantees in this adversarial setting are pessimistic, requiring the fraction of incorrectly matched seeds to be $o(1)$ (cf. Barak et al. (2018, Lemma 3.21) and Ding et al. (2021, Lemma 5)).

In this paper, we adopt a mathematically more tractable model introduced by Lubars and Srikant (2018), where the partially correct seeds are assumed to be generated independently from the graphs G_1 and G_2. More specifically, we use $\pi : [n] \to [n]$ to denote an initial mapping and generate π in the following way. For $\beta \in [0, 1)$, we assume that π is uniformly and randomly chosen from all the permutations $\sigma : [n] \to [n]$ such that $\sigma(u) = \pi^*(u)$ for exactly βn vertices. The benefit of this model is that π is independent of the graph G and the sampling processes that generate G_1 and G_2, and it is convenient for us to obtain theoretical results. For each seed $(u, \pi(u))$, if $\pi(u) = \pi^*(u)$, then $(u, \pi(u))$ is a correct seed; if $\pi(u) \neq \pi^*(u)$, then $(u, \pi(u))$ is an incorrect seed. Thus, only β fraction of the seeds are correct. Given G_1, G_2 and π, our goal is to find a mapping $\widetilde{\pi} : [n] \to [n]$ such that $\lim_{n\to\infty} \mathbb{P}\{\widetilde{\pi} = \pi^*\} = 1$.

Notation For any $n \in \mathbb{N}$, let $[n] = \{1, 2, \cdots, n\}$. We use standard asymptotic notation: for two positive sequences $\{a_n\}$ and $\{b_n\}$, we write $a_n = O(b_n)$ or $a_n \lesssim b_n$, if $a_n \leq Cb_n$ for some an absolute constant C and for all n; $a_n = \Omega(b_n)$ or $a_n \gtrsim b_n$, if $b_n = O(a_n)$; $a_n = \Theta(b_n)$ or $a_n \asymp b_n$, if $a_n = O(b_n)$ and $a_n = \Omega(b_n)$; $a_n = o(b_n)$ or $b_n = \omega(a_n)$, if $a_n/b_n \to 0$ as $n \to \infty$.

3. Algorithm Description

In this section, we present a general class of algorithms, shown in Algorithm 1, that we will use to recover π^*. Similar to Lubars and Srikant (2018), our algorithm also uses the notion of "witnesses". However, unlike Lubars and Srikant (2018), our algorithm leverages witnesses that are j-hop away. Given any graph G and two vertices u, v in G, we denote the length of the shortest path from u to v in G by $d^G(u, v)$. Then, for each vertex-pair (u, v), the seed $(w, \pi(w))$ becomes a j-hop witness for (u, v) if $d^{G_1}(u, w) = j$ and $d^{G_2}(v, \pi(w)) = j$.

We define the j-hop adjacency matrices $A_j \in \{0, 1\}^{n\times n}$ of G_1. Each element of A_j indicates whether a pair of vertices are j-hop neighbors in graph G_1, i.e., $A_j(u, v) = 1$ if $d^{G_1}(u, v) = j$ and $A_j(u, v) = 0$ otherwise. Similarly, let $B_j \in \{0, 1\}^{n\times n}$ denote the j-hop adjacency matrix of G_2. Equivalently express the seed mapping π by forming a permutation

matrix $\Pi \in \{0,1\}^{n\times n}$, where $\Pi(u,v) = 1$ if $\pi(u) = v$, and $\Pi(u,v) = 0$ otherwise. We can then count the number of j-hop witnesses for all vertex-pairs by computing $W_j = A_j \Pi B_j$, where the (u,v)-th entry of W_j is equal to the number of j-hop witnesses for the vertex-pair (u,v). This step has computational complexity same as matrix multiplication $O(n^\omega)$ with $2 \le \omega \le 2.373$ (Le Gall, 2014). As we have mentioned, a true pair tends to have more witnesses than a fake pair, and thus we want to find the vertex correspondence between the two graphs that maximizes the total number of witnesses. In other words, given a weighted bipartite graph G_m with the vertex set being a collection of all vertices in G_1 and G_2, the edges connecting every possible vertex-pairs, and weight of an edge defined as $w(u,v) = W_j(u,v)$, we want to find the matches in G_m with large weights. This linear assignment problem can be solved by the Hungarian algorithm in $O(n^3)$ time (Edmonds and Karp, 1972), which is computationally expensive for large values of n. To reduce computational complexity, we use Greedy Maximum Weight Matching (GMWM) with computational complexity $O(n^2 \log n)$. As we will show in Section 5, using GMWM is sufficient for finding the exact matching. GMWM first chooses the vertex-pair with the largest weight from all candidate vertex-pairs in G_m, removes all edges adjacent to the chosen vertex-pair, and then chooses the vertex-pair with the largest weight among the remaining candidate vertex-pairs, and so on. The total computational complexity of Algorithm 1 for any constant j is $O(n^\omega + n^2 \log n)$ for $2 \le \omega \le 2.373$.

When graphs are sufficiently sparse with average degree c, we can improve the time complexity of Algorithm 1 to $O(nc^{2j} + n^2 \log n)$ by computing the number of j-hop witnesses via neighborhood exploration. Moreover, we can further improve the scalability of the j-hop algorithm via parallel implementation. See Appendix B for details.

Algorithm 1 Graph Matching based on Counting j-hop Witnesses.

1: **Input:** G_1, G_2, π, j
2: Generate j-hop adjacency matrices A_j and B_j based on G_1 and G_2, and Π based on π;
3: **Output** $\widetilde{\pi} = \mathrm{GMWM}(W_j)$, where $W_j = A_j \Pi B_j$.

4. Main Results

In this section, we present the performance guarantees for the 1-hop and 2-hop algorithms.

Theorem 1 *If condition (3) holds and n is sufficiently large, then Algorithm 1 with $j = 1$ outputs $\widetilde{\pi}$ such that $\mathbb{P}\{\widetilde{\pi} = \pi^*\} \ge 1 - n^{-1}$.*

Comparing our condition (3) with the previous condition (2) in (Lubars and Srikant, 2018) as depicted in Fig. 1, we see that condition (3) requires significantly fewer correct seeds than condition (2) for dense graphs when $p = \Omega(\sqrt{\log n/n})$; thus, the 1-hop algorithm succeeds in exact recovery even when the fraction of correct seeds is significantly lower than the theoretical prediction in Lubars and Srikant (2018).

However, when $p = O(\sqrt{\log n/n})$, condition (3) still requires β to grow inversely proportional to np. As we have discussed, this is because when the graph is sparse, there are not enough 1-hop witnesses among the true pairs. Next, we show that by utilizing 2-hop

witnesses, our 2-hop algorithm succeeds in exact recovery with many fewer correct seeds in sparse graphs.

Theorem 2 *Suppose that* $np^2 \le \frac{1}{\log n}$ *and* $nps^2 \ge 128 \log n$. *If condition (4) holds and* n *is sufficiently large, then Algorithm 1 with* $j = 2$ *outputs* $\widetilde{\pi}$ *such that* $\mathbb{P}\{\widetilde{\pi} = \pi^*\} \ge 1 - n^{-1}$.

Our condition (4) is depicted as the blue curve in Fig. 1. At a high level, the three terms in (4) can be interpreted as follows:

- The first term $\beta \gtrsim \frac{\log n}{n^2p^2s^4}$ is to ensure that every true pair has more 2-hop witnesses contributed by the correct seeds than every fake pair. To see this, recall that there are $n\beta$ correct seeds. Since a true pair has about nps^2 1-hop common neighbors, each correct seed becomes a 2-hop witness for a true pair with probability about $nps^2 \cdot ps^2 = np^2s^4$. In contrast, for a fake pair, assuming independence of the 1-hop neighborhoods of the two vertices corresponding to the fake pair, the probability that each correct seed becomes a 2-hop witness for the fake pair is roughly $(nps)^2 \cdot (ps)^2 = n^2p^4s^4$. Hence, to ensure that a true pair has more 2-hop witnesses from the correct seeds than a fake pair, we at least need the difference between their means, i.e., $(n\beta)np^2s^4 - (n\beta)n^2p^4s^4$, to be positive. This is guaranteed by $np^2 \lesssim 1$, in which case the mean difference can be approximated by $\beta n^2p^2s^4$. However, due to randomness, we also need this mean difference to be larger than the standard deviation, which is on the order of $\sqrt{\beta n^2p^2s^4}$. This is guaranteed by $\beta \gtrsim \frac{1}{n^2p^2s^4}$. Adding the extra $\log n$ factor ensures that the above claim holds for every pair with high probability. This condition coincides with the seed requirement established in Mossel and Xu (2019) when the seeds are all correct;

- The second term $\beta \gtrsim \sqrt{\frac{\log n}{ns^4}}$ is due to the negative impact of the incorrect seeds. Note that there are $n(1-\beta)$ incorrect seeds, and each seed becomes a 2-hop witness for both a true pair and a fake pair with probability about n^2p^4. Although this contributes the same mean number of witnesses to both a true pair and fake pair, its randomness may contribute more to a fake pair than to a true pair. Thus, we need its standard deviation (on the order of $\sqrt{n(1-\beta)n^2p^4s^4}$) to be less than the mean difference $\beta n^2p^2s^4$ estimated in the first bullet. This is guaranteed by $\beta \gtrsim \sqrt{\frac{1}{ns^4}}$. Again, adding the $\log n$ factor ensures that the above claim holds for every pair with high probability.

- The third term $\beta \gtrsim \sqrt{np^3(1-s)\log n}$ is also caused by the incorrect seeds. However, the reason is more subtle than the second bullet, and is due to the fluctuation of the number of 1-hop neighbors of a true pair. Note that if $s = 1$, then, in both G_1 and G_2, the two vertices corresponding to a true pair have the same set of 1-hop neighbors, and thus the aforementioned fluctuation disappears. If instead $s < 1$, then the vertices corresponding to a true pair will have a different set of 1-hop neighbors in G_1 and G_2. This variation makes it even harder to distinguish the true pairs from the fake pairs based on the number of 2-hop witnesses as p increases, which gives to the condition $\beta \gtrsim \sqrt{np^3(1-s)\log n}$. Please refer to (30) in Section 5.2.4 for detailed derivation.

As a consequence, the blue curve has two branches: the top branch holds for $s < 1$ and the bottom one holds for $s = 1$.

As readers can see, in the latter two cases, our condition captures the new effect of the incorrect seeds and thus are significantly different from the theoretical results in Mossel and Xu (2019) where the seeds are all correct. Please refer to Remark 5.2.3 in Section 5.2 for more detailed discussions.

Pictorially, the three terms lead to the three segments in the blue curve in Fig. 1:

- When $p \lesssim \left(\frac{\log n}{n^3}\right)^{\frac{1}{4}}$, the first term of (4) dominates, as the graphs are so sparse that the 2-hop witnesses contributed by the incorrect seeds become negligible;
- When $\left(\frac{\log n}{n^3}\right)^{\frac{1}{4}} \lesssim p \lesssim n^{-\frac{2}{3}}$, the second term dominates, as the influence of the incorrect seeds cannot be ignored. In this case, the bottleneck for the success of the 2-hop algorithm is due to the statistical fluctuation of the 2-hop witnesses contributed by the incorrect seeds;
- When $n^{-\frac{2}{3}} \lesssim p \lesssim (n \log n)^{-\frac{1}{2}}$ and $s < 1$, the third term dominates, as the fluctuation of the 1-hop neighborhood sizes of the true pair increases with p and becomes the new bottleneck.

From Fig. 1, we observe that our 2-hop algorithm requires substantially fewer correct seeds to succeed than the 1-hop algorithm when the graphs are sparse. Moreover, our 2-hop algorithm is comparable to the NoisySeeds algorithm for very sparse graphs when $p \ll n^{-\frac{5}{6}}$, but continues to perform well over a much wide range of graph sparsity up to $p \lesssim (n \log n)^{-1/2}$.

Next, we present the necessary condition for the exact recovery with partially-correct seeds and compare it to our achievable results.

Theorem 3 *If*

$$nps^2 - \log n = O(1),$$

then any algorithm outputs $\widetilde{\pi} \neq \pi^*$ with at least a probability of $\Omega((1-\beta)^3)$.

The intuition behind Theorem 3 is as follows. Let $G_1^{\pi^*}$ denote the graph obtained by relabeling every vertex i in G_1 by $\pi^*(i)$. In this way, any two vertices corresponding to a true pair have the same label in $G_1^{\pi^*}$ and G_2. Denote the intersection graph by $G_1^{\pi^*} \wedge G_2$ which includes the common edges in both $G_1^{\pi^*}$ and G_2. The main idea of Theorem 3 is that it is impossible to recover the true matching of any isolated vertex in $G_1^{\pi^*} \wedge G_2$ that is incorrectly seeded. Therefore, we need $nps^2 - \log n \to +\infty$ so that there is no isolated vertex in $G_1^{\pi^*} \wedge G_2$. Detailed proof of Theorem 3 is provided in Appendix F.

Note that the above information-theoretic limit of exact recovery with partially-correct seeds coincides with that without seeds (Cullina and Kiyavash, 2016, 2017; Wu et al., 2021). Thus, seeds do not improve the information-theoretic limit for exact recovery compared to that without seeds. However, to our best knowledge, achieving these information-theoretic limits requires algorithms with super-polynomial time. Seeds do help in designing polynomial-time algorithms as our polynomial-time seeded matching algorithms can tolerate lower values of s than polynomial-time seedless matching algorithms (see discussions in Section 1).

Further, we acknowledge that there is a gap between the sufficient conditions for our algorithms and the above information-theoretic limit of exact recovery. Recall that the 1-hop algorithm requires a sufficient condition $nps^2 \geq \frac{45 \log n}{(1-p)^2 \beta}$, and our 2-hop algorithm requires a sufficient condition $nps^2 \geq 128 \log n$. It remains open whether the information-theoretic limit can be achieved in polynomial time.

5. Analysis

In this section, we explain the intuition and sketch the proofs for Theorem 1 and Theorem 2. In the analysis, we assume without loss of generality that the true mapping π^* is the identity mapping, i.e., $\pi^*(i) = i$.

5.1 Intuition and Proof of Theorem 1

To understand the intuition behind Theorem 1 and why it provides a better result than Lubars and Srikant (2018), recall that the 1-hop algorithm will succeed (in recovering π^*) if the number of 1-hop witnesses for any true pair is larger than the number of 1-hop witnesses for any fake pair. For any correct seed, it is a 1-hop witness for a true pair with probability ps^2 and is a 1-hop witness for a fake pair with probability $p^2 s^2$. In contrast, for any incorrect seed, it is a 1-hop witness with probability $p^2 s^2$ for both true pairs and fake pairs. Since there are $n\beta$ seeds that are correct, it follows that

$$W_1(u,v) \dot{\sim} \begin{cases} \text{Binom}\left(n\beta, ps^2\right) + \text{Binom}\left(n(1-\beta), p^2 s^2\right) & \text{if } u = v, \quad (5a) \\ \text{Binom}(n, p^2 s^2) & \text{if } u \neq v. \quad (5b) \end{cases}$$

where $\dot{\sim}$ denotes "approximately distributed".

For fake pair $u \neq v$, using Bernstein's inequality given in Theorem 15 in Appendix C, we show that $W_1(u,v)$ is upper bounded by $np^2 s^2 + O(\sqrt{np^2 s^2 \log n}) + O(\log n)$ with high probability. More precisely, we have the following lemma, with the proof deferred to Appendix D.1.

Lemma 4 *For any two vertices $u, v \in [n]$ with $u \neq v$ and sufficiently large n, the following holds*

$$\mathbb{P}\left\{W_1(u,v) < \psi_{\max}\right\} \geq 1 - n^{-\frac{7}{2}}, \quad (6)$$

where $\psi_{\max} = np^2 s^2 + \sqrt{7np^2 s^2 \log n} + \frac{7}{3} \log n + 2$.

For true pair $u = v$, the first binomial distribution in (5a) can be lower bounded by $n\beta ps^2 - O(\sqrt{n\beta ps^2 \log n}) - O(\log n)$ with high probability using Bernstein's inequality. However, the second Binomial distribution in (5a) is not precise because the events that each incorrect seed becomes a witness for a true pair are dependent on each other, as we discussed in Section 1.1. We address this dependency issue using the concentration inequality for dependent random variables (Janson, 2004), and get the following lower bound on the number of 1-hop witnesses for the true pairs.

Lemma 5 *For any vertex $u \in [n]$ and sufficiently large n, the following holds*

$$\mathbb{P}\left\{W_1(u,u) > x_{\min} + y_{\min}\right\} \geq 1 - n^{-\frac{7}{3}}, \quad (7)$$

where

$$x_{\min} = (n\beta - 1)ps^2 - \sqrt{5n\beta ps^2 \log n} - \frac{5}{3}\log n,$$
$$y_{\min} = (n(1-\beta) - 2)p^2 s^2 - 5\sqrt{np^2 s^2 \log n} - \frac{25}{3}\log n.$$

Proof [Proof of Lemma 5] Recall that A_1 and B_1 are the adjacency matrix for G_1 and G_2, respectively. Let $F \triangleq \{i : \pi(i) = i\}$ denote the set of fixed points of π. Then F corresponds to the set of correct seeds with $|F| = n\beta$ (recall that we assume the true matching π^* to be the identity mapping). By the definition of 1-hop witness, we have

$$\begin{aligned} W_1(u,u) &= \sum_{i\in F} A_1(u,i)B_1(u,i) + \sum_{i\in[n]\setminus F} A_1(u,i)B_1(u,\pi(i)) \\ &= \sum_{i\in F\setminus\{u\}} A_1(u,i)B_1(u,i) + \sum_{i\in[n]\setminus(F\cup\{u,\pi^{-1}(u)\})} A_1(u,i)B_1(u,\pi(i)), \end{aligned} \tag{8}$$

where the second equality holds because $A_1(u,u) = B_1(u,u) = 0$. Let $X_i \triangleq A_1(u,i)B_1(u,i)$ for $i \in F \setminus \{u\}$ and $Y_i \triangleq A_1(u,i)B_1(u,\pi(i))$ for $i \in [n] \setminus (F \cup \{u, \pi^{-1}(u)\})$.

For all $i \in F \setminus \{u\}$, $X_i \overset{\text{i.i.d.}}{\sim} \text{Bern}(ps^2)$. It follows that

$$\mathbb{P}\left\{\sum_{i\in F} X_i \le x_{\min}\right\} \le \mathbb{P}\left\{\text{Binom}(n\beta - 1, ps^2) \le x_{\min}\right\} \le n^{-\frac{5}{2}}, \tag{9}$$

where that last inequality follows from Bernstein's inequality given in Theorem 15 with $\gamma = \frac{5}{2}\log n$ and $K = 1$.

For all $i \in [n] \setminus \big(F \cup \{u, \pi^{-1}(u)\}\big)$, $Y_i \sim \text{Bern}(p^2 s^2)$. However, Y_i's are dependent and thus we cannot directly apply Bernstein's inequality. To see this, $A_1(u,i)$ and $B_1(u,i)$ are correlated, but $\{A_1(u,i), B_1(u,i)\}$ are independent across different (u,i). Let $S_i = \{\{u,i\}, \{u,\pi(i)\}\}$. Thus, Y_i only depends on the set S_i of entries of A_1 and B_1. Since $S_i \cap S_{i'} \neq \emptyset$ if and only if $i' = \pi(i)$ or $i' = \pi^{-1}(i)$, it follows that Y_i is dependent on $Y_{i'}$ if and only if $i' = \pi(i)$ or $i' = \pi^{-1}(i)$. Then we can construct a dependency graph Γ for $\{Y_i\}$, where the maximum degree of Γ, $\Delta(\Gamma)$, equals to two. Hence, applying the concentration inequality for the sum of dependent random variables given in Theorem 18 with $\gamma = \frac{8}{3}\log n$ and $K = 1$ yields that

$$\mathbb{P}\left\{\sum_{i\in[n]\setminus(F\cup\{u,\pi^{-1}(u)\})} Y_i \le y_{\min}\right\} \le n^{-\frac{8}{3}}. \tag{10}$$

Finally, combining (8), (9) and (10) and applying union bound yields the desired conclusion (7). ■

Combining Lemma 4 and Lemma 5, for the 1-hop algorithm to succeed, it suffices to ensure that $x_{\min} + y_{\min} \ge \psi_{\max}$. Note that

$$x_{\min} + y_{\min} - \psi_{\max} \ge 0$$

$$\Leftarrow \frac{1}{3}n\beta p(1-p)s^2 \geq \sqrt{5n\beta ps^2 \log n}, \text{ and } \frac{1}{3}n\beta p(1-p)s^2 \geq (5+\sqrt{7})\sqrt{np^2s^2\log n}, \text{ and } \frac{1}{3}n\beta p(1-p)s^2 \geq \frac{37}{3}\log n + 2 + ps^2 + 2p^2s^2, \tag{11}$$

which is implied by condition (3) in Theorem 1. Thus, by taking the union bound over (6) and (7), we complete the proof of Theorem 1. Please refer to Appendix D.2 for details. The above argument suggests that the sufficient condition (3) is also close to necessary (differing from the necessary condition by a constant factor) for the 1-hop algorithm to succeed, which is confirmed by our simulation results in Appendix A.

5.2 Intuition and Proof of Theorem 2

We next explain the intuition and sketch the proof of Theorem 2 when $np^2 \leq \frac{1}{\log n}$ and $nps^2 \geq 128\log n$.

We start by explaining why Theorem 2 requires $np^2 \leq \frac{1}{\log n}$ and $nps^2 \geq 128\log n$. First, note that the intersection graph $G_1^{\pi^*} \wedge G_2$ (which includes edges appearing in both G_1 and G_2) is an Erdős-Rényi random graph with average degree $(n-1)ps^2$. Thus, we need $nps^2 \geq 128\log n$ so that there is no isolated vertex in $G_1^{\pi^*} \wedge G_2$. Otherwise, it is impossible to match the isolated vertices and reach the goal of perfect matching (Cullina and Kiyavash, 2016, 2017; Wu et al., 2021). Moreover, we will use this condition in Section 5.2.1 to ensure that the number of 1-hop neighbors is concentrated. Second, the condition $np^2 \leq 1/\log n$ ensures that the graph is not too dense so that the true pair is expected to have more 2-hop witnesses than the fake pair. Please see later in (15) how this condition arises.

Then, analogous to the 1-hop algorithm, we derive the condition on β by comparing the number of 2-hop witnesses for true pairs and for fake pairs. However, the dependency issue is more severe here when we bound the number of 2-hop witnesses. Specifically, in the analysis of Lemma 5, the event that an incorrect seed becomes a 1-hop witness for a true pair is dependent on that of at most two other incorrect seeds. However, for 2-hop witnesses, any two seeds could be dependent through the 1-hop neighborhoods of the candidate vertex-pair (see Fig. 3 for an example). Thus, directly using the concentration inequality in Janson (2004) will lead to a poor bound. To address this new difficulty, we will condition on the 1-hop neighborhoods first. After this conditioning, the remaining dependency becomes more manageable, which is handled by either classifying the seeds or by applying the concentration inequality in Janson (2004) again.

5.2.1 Bound on the 1-hop Neighbors

In order to condition on the typical sizes of the 1-hop neighborhoods, we first bound the number of 1-hop neighbors. For any vertex u in graph G, we use $N^G(u)$ to denote the set of 1-hop neighbors of u in G, i.e., $N^G(u) = \left\{v \in G : d^G(u,v) = 1\right\}$. For any two vertices $u, v \in [n]$, let $C(u,v)$ denote the set of 1-hop "common" neighbors of u and v across G_1 and G_2, i.e., $C(u,v) = N^{G_1}(u) \cap N^{G_2}(v)$. For ease of notation, let

$$d_u = \left|N^{G_0}(u)\right|, \quad a_u = \left|N^{G_1}(u)\right|, \quad b_v = \left|N^{G_2}(v)\right|, \quad c_{uv} = \left|C(u,v)\right|.$$

By definition, we have $a_u, b_v \sim \text{Binom}(n-1, ps)$, $c_{uu} \sim \text{Binom}(n-1, ps^2)$, and $c_{uv} \sim \text{Binom}(n-1, p^2s^2)$ for $u \neq v$. Thus, by using concentration inequalities for binomial distri-

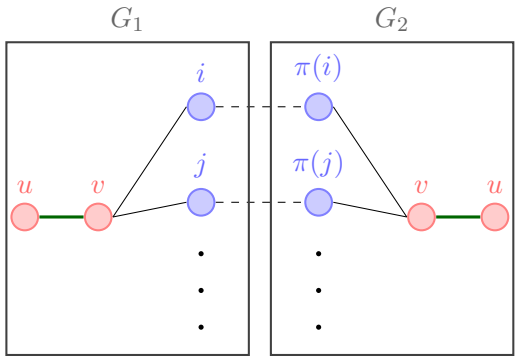


Figure 3: The events that any seed $(i, \pi(i))$ or $(j, \pi(j))$ becomes a 2-hop witness for (u, u) are dependent through the 1-hop neighbors of u. Specifically, knowing that $(i, \pi(i))$ is a 2-hop witness for (u, u) means that some vertex v is connected to u, which will influence the probability that any other seed $(j, \pi(j))$ becomes a 2-hop witness for (u, u). Thus, there exists dependency across any two seeds.

butions and letting

$$\epsilon = \sqrt{\frac{12 \log n}{(n-1)ps^2}} \leq \frac{1}{3}, \tag{12}$$

where the last inequality holds due to the assumption $nps^2 \geq 128 \log n$, We can show that with high probability, a_u and b_u are bounded by $(1 \pm \epsilon)nps$, c_{uu} is bounded by $(1 \pm \epsilon)nps^2$, and c_{uv} is upper bounded by $\psi_{\max}$ in Lemma 6 below. In particular, we arrive at the following lemma with the proof deferred to Appendix E.1.

Lemma 6 *Given any two vertices $u, v \in [n]$ with $u \neq v$, let R_{uv} denote the event such that the followings hold simultaneously:*

$$\begin{gathered}(1-\epsilon)(n-1)ps < a_u, a_v, b_u, b_v < (1+\epsilon)(n-1)ps, \\ (1-\epsilon)(n-1)ps^2 < c_{uu}, c_{vv} < (1+\epsilon)(n-1)ps^2, \\ c_{uv},\, W_1(v,u) < \psi_{\max},\end{gathered}$$

where $\psi_{\max} = np^2s^2 + \sqrt{7np^2s^2 \log n} + \frac{7}{3}\log n + 2$.

If $nps^2 \geq 128 \log n$, then for all sufficiently large n,

$$\mathbb{P}\{R_{uv}\} \geq 1 - n^{-\frac{7}{2}}. \tag{13}$$

5.2.2 Bound on the 2-hop Witnesses

In the sequel, we condition on the 1-hop neighborhoods of u and v such that event R_{uv} holds, and bound the 2-hop witnesses for both the true pairs and fake pairs. To compute the probability that a seed $(j, \pi(j))$ becomes a 2-hop witness for pair (u, v), we calculate the *joint probability* that j connects to some 1-hop neighbor of u in G_1 and $\pi(j)$ connects to some 1-hop neighbor of v in G_2. For any correct seed, it is a 2-hop witness for a true pair (u, u)

with probability[1] about $c_{uu}ps^2 + (a_u b_u - c_{uu})p^2 s^2$, where the first term is the dominating term, and is a 2-hop witness for a fake pair (u, v) with probability about $a_u b_v p^2 s^2$. In contrast, for any incorrect seed, it is a 2-hop witness for a true pair (u, u) with probability about $a_u b_u p^2 s^2$ and is a 2-hop witness for a fake pair (u, v) with probability about $a_u b_v p^2 s^2$. Thus we have

$$W_2(u,v) \dot{\sim} \begin{cases} \text{Binom}\left(n\beta, c_{uu}ps^2\right) + \text{Binom}\left(n(1-\beta), a_u b_u p^2 s^2\right) & \text{if } u = v, \quad (14a) \\ \text{Binom}\left(n, a_u b_v p^2 s^2\right) & \text{if } u \neq v. \quad (14b) \end{cases}$$

To ensure that the numbers of 2-hop witnesses are separated between true pairs and fake pairs, we need $\mathbb{E}\left[W_2(u,u)\right] \geq \mathbb{E}\left[W_2(u,v)\right]$ for $u \neq v$, which, in view of (14a) and (14b), $a_u, b_u, b_v \approx nps$, and $c_{uu} \approx nps^2$, amounts to

$$n\beta(nps^2)ps^2 + n(1-\beta)(nps)^2 p^2 s^2 - n(nps)^2 p^2 s^2 \geq 0 \Leftrightarrow np^2 \leq 1. \quad (15)$$

This shows that the 2-hop algorithm is only effective when the graphs are sufficiently sparse. For this reason, we assume $np^2 \leq 1/\log n$ so that (15) is satisfied.

For the 2-hop algorithm to be effective, we also need to consider the statistical fluctuation of $W_2(u,v)$. For true pair $u = v$, using Bernstein's inequality, $\text{Binom}(n\beta, c_{uu}ps^2)$ is lower bounded by $n\beta c_{uu}ps^2 - O(\sqrt{n\beta c_{uu}ps^2 \log n}) - O(\log n)$ with high probability. However, the second Binomial distribution in (14a) is not precise because the events that each incorrect seed becomes a 2-hop witness for a true pair are dependent on other incorrect seeds. Fortunately, similar to the proof of Lemma 5, we can deal with this dependency issue using the concentration inequality for dependent random variables (Janson, 2004). Thus, we can get the following lower bound on the number of 2-hop witnesses for the true pairs conditional on the 1-hop neighborhoods.

Lemma 7 *Given any two vertices $u, v \in [n]$ with $u \neq v$, we use Q_{uv} to collect all information of 1-hop neighborhood of u and v, i.e.,*

$$Q_{uv} = \left\{N^{G_1}(u), N^{G_2}(u), N^{G_1}(v), N^{G_2}(v)\right\}.$$

If n is sufficiently large and $nps^2 \geq 128 \log n$, then

$$\mathbb{P}\left\{W_2(u,u) \leq l_{\min} + m_{\min} \mid Q_{uv}\right\} \cdot \mathbb{1}(R_{uv}) \leq n^{-\frac{7}{2}}, \quad (16)$$

where

$$l_{\min} = \frac{7}{24}(1-\delta_1)\beta n^2 p^2 s^4 - \sqrt{\frac{35}{16}\beta n^2 p^2 s^4 \log n} - \frac{5}{2}\log n, \quad (17)$$

$$\begin{aligned} m_{\min} =& n(1-\beta)\left(1-(1-ps)^{a_{u\setminus v}}\right)\left(1-(1-ps)^{b_{u\setminus v}}\right) \\ & - 21n^3p^5s^5 - \frac{15}{2}\sqrt{\frac{3}{2}n^3p^4s^4\log n} - \frac{25}{2}\log n, \end{aligned} \quad (18)$$

with $\delta_1 = \frac{6ps}{\beta}$, $a_{u\setminus v} = \left|N^{G_1}(u) \setminus \{v\}\right|$, and $b_{u\setminus v} = \left|N^{G_2}(u) \setminus \{v\}\right|$.

1. Among all $a_u b_u$ possible cases that a correct seed connects to 1-hop neighbors of (u,u), there are only c_{uu} cases that the correct seed connects to the common 1-hop neighbors of (u,v). Thus, we have c_{uu} in the first term and $(a_u b_u - c_{uu})$ in the second term.

Remark 8 *Note that $l_{\min}$ is contributed by the correct seeds and $m_{\min}$ is contributed by the incorrect seeds. Specifically, conditional on the 1-hop neighbors, a correct seed becomes a 2-hop witness for the true pair (u,u) with probability about $c_{uu}ps^2 \approx np^2s^4$. Multiplying by $n\beta$ gives an expression close to the first term of $l_{\min}$. Similarly, an incorrect seed becomes a 2-hop witness for the true pair (u,u) with probability about $(1-(1-ps)^{a_{u\setminus v}})\left(1-(1-ps)^{b_{u\setminus v}}\right)$. Multiplying by $n(1-\beta)$ gives the first term of $m_{\min}$. In summary, the first term in $l_{\min}$ and $m_{\min}$ is a lower bound of the expectation, and the rest of the terms are due to the tail bounds.*

Due to the conditioning of 1-hop neighborhoods, we exclude seeds that are 1-hop neighbors of u when bounding $W_2(u,u)$, giving rise to the additional δ_1 and $21n^3p^5s^5$ terms in Lemma 7. Please refer to Appendix E.2 for the proof.

For the fake pair $u \neq v$, we have the following upper bound on the number of 2-hop witnesses for the fake pairs conditional on the 1-hop neighborhoods.

Lemma 9 *For any two vertices $u, v \in [n]$ with $u \neq v$, if $nps^2 \geq 128\log n$, then for all sufficiently large n,*

$$\mathbb{P}\left\{W_2(u,v) \geq x_{\max} + y_{\max} + 2z_{\max} + \psi_{\max} + 28\log n \mid Q_{uv}\right\} \cdot \mathbb{1}\left(R_{uv}\right) \leq n^{-\frac{7}{2}}. \tag{19}$$

where

$$x_{\max} = 2n\beta\left(\psi_{\max}ps^2 + \frac{9}{4}n^2p^4s^4\right), \tag{20}$$

$$y_{\max} = n(1-\beta)\left(1-(1-ps)^{a_{u\setminus v}}\right)\left(1-(1-ps)^{b_{v\setminus u}}\right) + n^2p^3s^3 + \frac{5}{2}\sqrt{15n^3p^4s^4\log n}, \tag{21}$$

$$z_{\max} = \frac{9}{2}n^2p^3s^3,$$

$$\psi_{\max} = np^2s^2 + \sqrt{7np^2s^2\log n} + \frac{7}{3}\log n + 2. \tag{22}$$

Remark 10 *Note that if u and v are connected in G_1, the conditioning on Q_{uv} changes the probability that the seed $(j,\pi(j))$ with $j \in N^{G_1}(v)$ becomes a 2-hop witness for (u,v). Thus, we have to divide the seeds into several types depending on whether $j \in N^{G_1}(v)$ or $\pi(j) \in N^{G_2}(u)$, and consider their contribution to the number of 2-hop witnesses separately:*

1) *$x_{\max} + y_{\max}$ is the major term in (19) and is contributed by the seeds such that $j \notin N^{G_1}(v) \cup \pi^{-1}\left(N^{G_2}(u)\right)$ (see Fig. 4(a) for an example). In the analysis, we further divide such seeds into two categories, where $x_{\max}$ is contributed by the correct seeds, and $y_{\max}$ is contributed by the incorrect seeds. Specifically, conditional on the 1-hop neighbors, a correct seed becomes a 2-hop witness for the fake pair (u,v) either when the two vertices of the seed connect to different 1-hop neighbors of u and v, respectively, or when they connect to a common 1-hop neighbor of u and v. Thus, the conditional probability of such event is about $c_{uv}ps^2 + a_ub_vp^2s^2$. According to Lemma 6, $\psi_{\max}$ is an upper bound estimate of c_{uv}, and both a_u and b_v are approximately nps. Therefore, the above conditional probability can be approximately estimated as $\psi_{\max}ps^2 + n^2p^4s^4$. Multiplying by $n\beta$ gives an expression close to $x_{\max}$.*

Similarly, an incorrect seed becomes a 2-hop witness for the fake pair (u, v) with probability about $(1-(1-ps)^{a_{u\setminus v}})\left(1-(1-ps)^{b_{v\setminus u}}\right)$. Multiplying by $n(1-\beta)$ gives the first term of $y_{\max}$. In summary, the first term in $x_{\max}$ and $y_{\max}$ is an upper bound of the expectation, and the rest of the terms are due to the tail bounds.

2) *One multiple of $z_{\max}$ in (19) is contributed by the seeds such that $j \in N^{G_1}(v) \setminus \pi^{-1}\left(N^{G_2}(u)\right)$ (see Fig. 4(b) for an example). To see this, note that there are roughly nps such seeds $(j, \pi(j))$. If u and v are connected in G_1, then j must be a 2-hop neighbor of u, i.e., $A_2(u, j) = 1$. On the other hand, the probability that $\pi(j)$ becomes a 2-hop neighbor of v is approximately np^2s^2. Thus, the expected number of 2-hop witnesses contributed by this type of seeds is approximately $n^2p^3s^3$. The other multiple of $z_{\max}$ in (19) is for the opposite case: it is contributed by the seeds such that $j \in \pi^{-1}\left(N^{G_2}(u)\right) \setminus N^{G_1}(v)$.*

3) *The term $\psi_{\max}$ in (19) is contributed by the seeds such that $j \in N^{G_1}(v) \cap \pi^{-1}\left(N^{G_2}(u)\right)$ (see Fig. 4(c) for an example). In this case, $(j, \pi(j))$ becomes a 1-hop witness for (v, u). Since $W_1(v, u) < \psi_{\max}$ according to Lemma 6, there are at most $\psi_{\max}$ such seeds.*

4) *The term $28 \log n$ in (19) comes from the sub-exponential tail bounds when applying concentration inequalities.*

Please refer to Appendix E.3 for the proof.

5.2.3 Derivation of a Sub-optimal Version of Condition (4)

By combining Lemma 7 and Lemma 9, we are ready to derive a sufficient (but not tight) condition for the success of the 2-hop algorithm. First, analogous to the proof of Theorem 1, for the 2-hop algorithm to succeed, it suffices that

$$\min_u W_2(u, u) > \max_{u \neq v} W_2(u, v). \tag{23}$$

Then by combining Lemma 7 and Lemma 9, (23) is guaranteed when

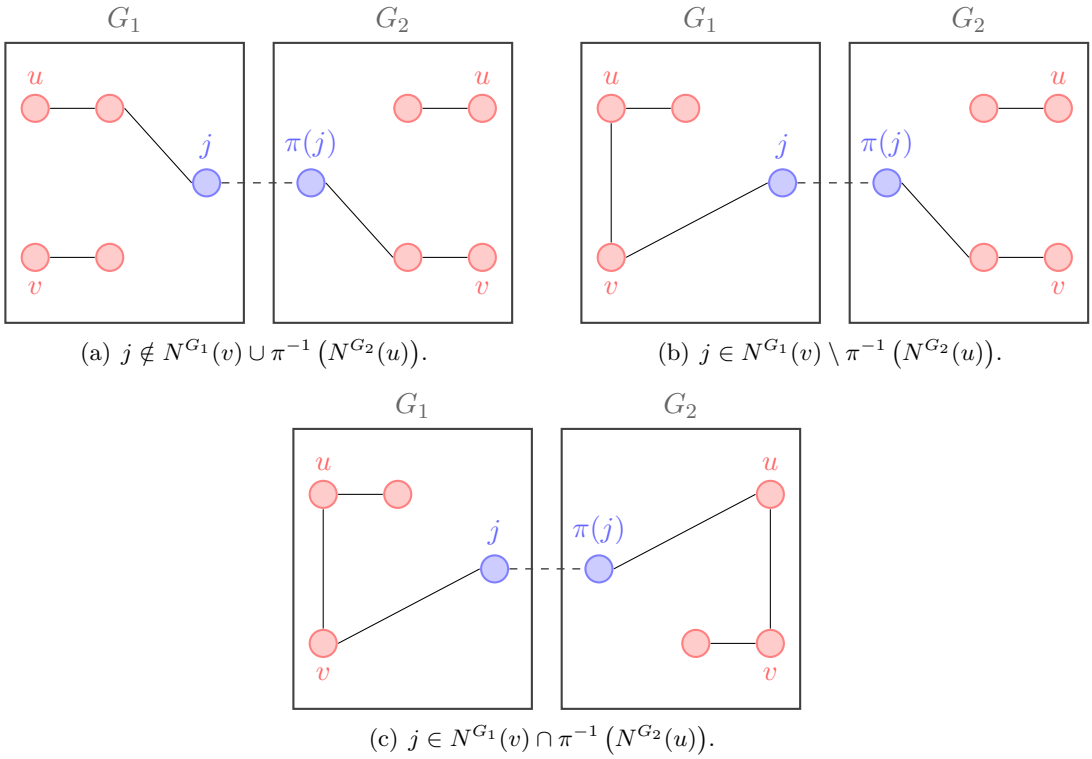
$$l_{\min} + m_{\min} \geq x_{\max} + y_{\max} + 2z_{\max} + \psi_{\max} + 28 \log n. \tag{24}$$

Finally, to ensure (24) is satisfied when $np^2 \leq \frac{1}{\log n}$ and $nps^2 \geq 128 \log n$, we arrive at the following sufficient condition:

$$\beta \gtrsim \max\left\{\frac{\log n}{n^2p^2s^4}, \sqrt{\frac{\log n}{ns^4}}, \sqrt{\frac{np^3 \log n}{s}}\right\}. \tag{25}$$

Note that condition (25) is similar to condition (4) except for the third term. It is instructive to see how (25) implies (24):

- When $\beta \gtrsim \frac{\log n}{n^2p^2s^4}$, $\beta \gtrsim \sqrt{\frac{\log n}{ns^4}}$, and $np^2 \leq \frac{1}{\log n}$, we have from (17) that $l_{\min} \geq c \cdot \beta n^2p^2s^4$ for some constant c. In other words, the true pair should have sufficiently many 2-hop witnesses from the correct seeds.



(a) $j \notin N^{G_1}(v) \cup \pi^{-1}\left(N^{G_2}(u)\right)$. (b) $j \in N^{G_1}(v) \setminus \pi^{-1}\left(N^{G_2}(u)\right)$.

(c) $j \in N^{G_1}(v) \cap \pi^{-1}\left(N^{G_2}(u)\right)$.

Figure 4: We divide the seeds into several types based on whether $j \in N^{G_1}(v)$ or $\pi(j) \in N^{G_2}(u)$.

- When $np^2 \le \frac{1}{\log n}$ and $nps^2 \ge 128 \log n$, we have from (20) that $x_{\max} \lesssim \beta nps^2 \log n \le \frac{c}{3}\beta n^2 p^2 s^4$, ensuring that the fake pairs have fewer 2-hop witnesses from the correct seeds than the true pairs.
- For the 2-hop witnesses from the incorrect seeds, we have from (18) and (21) that

$$m_{\min} \approx n(1-\beta)a_u b_u p^2 s^2 - \Delta$$
$$y_{\max} \approx n(1-\beta)a_u b_v p^2 s^2 + \Delta,$$

  where $\Delta = O(\sqrt{n^3 p^4 s^4 \log n}) + O(\log n)$ captures the statistical deviation.

  - When $\beta \gtrsim \frac{\log n}{n^2 p^2 s^4}$ and $\beta \gtrsim \sqrt{\frac{\log n}{ns^4}}$, we have $\Delta \le \frac{c}{3}\beta n^2 p^2 s^4$.
  - When $\beta \gtrsim \sqrt{\frac{np^3 \log n}{s}}$, in view of $a_u \lesssim nps$ and $b_v - b_u \lesssim \sqrt{nps \log n}$ (the latter one is due to the fluctuation of the 1-hop neighborhood sizes), we have that

$$n(1-\beta)a_u b_v p^2 s^2 - n(1-\beta)a_u b_u p^2 s^2 \lesssim n(1-\beta)nps\sqrt{nps \log n} p^2 s^2 \le \frac{c}{3}\beta n^2 p^2 s^4. \tag{26}$$

The above two claims together ensure that $y_{\max} - m_{\min}$, i.e., the difference between the true pairs and fake pairs of the 2-hop witnesses from the incorrect seeds, is dominated by $\frac{2c}{3}\beta n^2 p^4 s^4$.

- Finally, when $np^2 \le \frac{1}{\log n}$, $2z_{\max} + \psi_{\max} + 28\log n \lesssim \Delta$ and hence is negligible.

In sum, if condition (25) holds, then with high probability (23) is satisfied and thus the 2-hop algorithm exactly recovers the true vertex mapping π^*.

5.2.4 Derivation of the Tight Condition (4)

Unfortunately, condition (25) does not completely coincide with the desired condition (4). This is because the criteria (23) that we used for GMWM to succeed is too strict. Indeed, the GMWM algorithm may succeed even when (23) does not hold. For example, consider the case in Fig. 5 when a_u and b_u are both small, while b_v is large . Then, $W_2(u,v)$ may be larger than $W_2(u,u)$ and hence (4) is not satisfied. However, since $N^{G_1}(v)$ and $N^{G_2}(v)$ are expected to overlap significantly, when b_v is large, a_v is also likely to be large. Hence $W_2(v,v)$ is likely to be larger than $W_2(u,v)$. Thus, GMWM will still select the true pair (v,v) and eliminate the fake pair (u,v). From the above example, we can see that, for the 2-hop algorithm to succeed, it is sufficient to satisfy the following new criteria:

$$W_2(u,v) < W_2(u,u) \text{ or } W_2(u,v) < W_2(v,v), \quad \forall u \neq v. \tag{27}$$

Next, we show that under condition (4), with high probability the new criteria (27) is

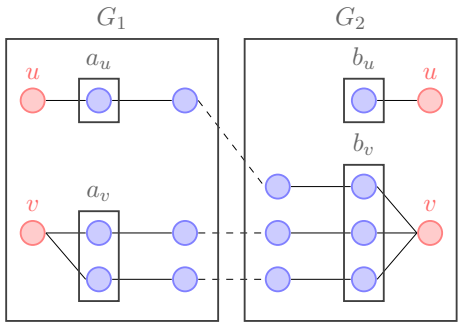


Figure 5: The 2-hop algorithm with GMWM still selects the true pair (v,v) and eliminate the fake pair (u,v) when $W_2(u,v) > W_2(u,u)$ but $W_2(v,v) > W_2(u,v)$.

satisfied and hence the 2-hop algorithm succeeds. Since $N^{G_1}(u)$ and $N^{G_2}(u)$ are both generated by sampling with probability s from $N^{G_0}(u)$ in the parent graph G_0, we have $a_u, b_u \sim \text{Binom}(d_u, s)$. Similarly, $a_v, b_v \sim \text{Binom}(d_v, s)$. Therefore, if $d_u \le d_v$, we expect $a_u - a_v$ not to be too large. If instead $d_u > d_v$, we expect $b_v - b_u$ not to be too large. More precisely, we have the following lemma, with the proof deferred to Appendix E.4.

Lemma 11 *Given any $u, v \in [n]$, let T_{uv} denote the event:*

$$T_{uv} = \{a_u - a_v \le \tau\} \cup \{b_v - b_u \le \tau\}, \tag{28}$$

where

$$\tau \triangleq 2\sqrt{10nps(1-s)\log n} + 5\log n. \tag{29}$$

If n is sufficiently large and $nps^2 \geq 128\log n$, then

$$\mathbb{P}(T_{uv}) \geq 1 - n^{-\frac{7}{2}}.$$

Next we show the new criteria (27) is satisfied by separately considering two cases: $b_v - b_u \leq \tau$ and $a_u - a_v \leq \tau$. We first consider the case $b_v - b_u \leq \tau$. When $\beta \gtrsim \sqrt{\frac{np^3(1-s)\log n}{s}}$, $\beta \gtrsim \sqrt{\frac{\log n}{ns^4}}$ and $np^2 \leq \frac{1}{\log n}$, in view of $a_u \lesssim nps$, we can get a tighter upper bound to the left hand side of (26):

$$\begin{aligned}
&n(1-\beta)a_u b_v p^2 s^2 - n(1-\beta)a_u b_u p^2 s^2 \\
\lesssim& n(1-\beta)nps\left(\sqrt{nps(1-s)\log n} + \log n\right)p^2 s^2 \\
\overset{(a)}{\leq}& n^2 p^3 s^3 \sqrt{nps(1-s)\log n} + n^2 p^2 s^3 \sqrt{\frac{\log n}{n}} \overset{(b)}{\leq} \frac{c}{3}\beta n^2 p^2 s^4,
\end{aligned} \tag{30}$$

where the inequality (a) holds due to $p \leq \sqrt{1/(n\log n)}$; the inequality (b) is guaranteed by the last two terms in condition (4).

To be more precise, the following lemma combined with Lemma 7 and Lemma 9 ensures that $W_2(u,u) > W_2(u,v)$ with high probability.

Lemma 12 *Given any two vertices $u, v \in [n]$ with $u \neq v$, if R_{uv} occurs, $b_v - b_u \leq \tau$, $nps^2 \geq 128\log n$, $np^2 \leq \frac{1}{\log n}$, and condition (4) holds, then for sufficiently large n,*

$$l_{\min} + m_{\min} \geq x_{\max} + y_{\max} + 2z_{\max} + \psi_{\max} + 28\log n.$$

where $l_{\min}$, $m_{\min}$, $x_{\max}$, $y_{\max}$, $z_{\max}$ and $\psi_{\max}$ are given in Lemma 7 and Lemma 9.

Please refer to Appendix E.5 for details.

For the other case that $a_u - a_v \leq \tau$, we instead bound $W_2(v,v)$ from below analogous to Lemma 7, and prove that $W_2(v,v) > W_2(u,v)$ with high probability analogous to Lemma 12.

Thus, by combining the two cases and applying union bound, we ensure that with high probability the new criteria (27) is satisfied and hence the GMWM outputs the true matching, completing the proof of Theorem 2. Please refer to Appendix E.6 for details. Note that, when $1 - s = o(1)$, condition (4) given by the new criteria (27) requires a smaller β than (25) given by the old criteria (23).

6. Numerical experiments

In this section, we present numerical studies, comparing the performance of our 2-hop algorithm to the 1-hop algorithm (Lubars and Srikant, 2018) and the NoisySeeds algorithm (Kazemi et al., 2015), which are the state-of-the-art for graph matching with imperfect seeds. Additional numerical studies to verify our theoretical results are deferred to Appendix A. In all our experiments except for the last one on the computer vision data set in Section 6.2.3,

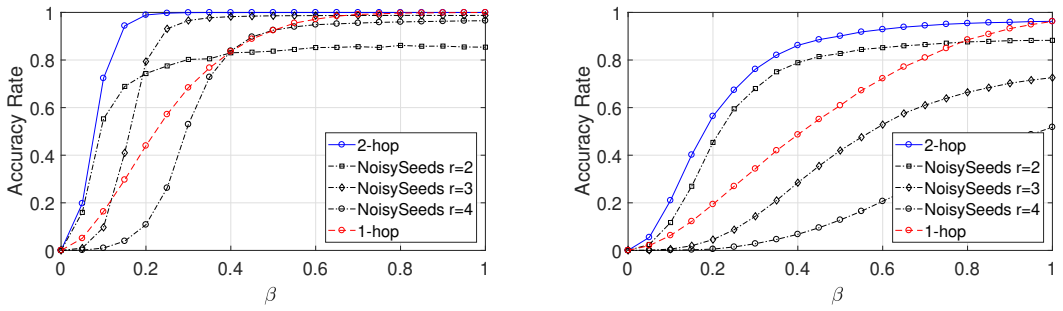
the initial seeded mappings are constructed in the same way as our model given in Section 2, i.e., the initial mappings are uniformly chosen at random with a given number of correctly matched pairs. To be more precise, we choose βn correct seeds among all true pairs uniformly, and then we permute the rest of the vertices uniformly at random such that they all form incorrect seeds. In contrast, in the computer vision experiment in Section 6.2.3, the initial seeded mapping is from the output of a seedless matching algorithm. The computational environment is MATLAB R2017a on a standard PC with 2.4 GHz CPU and 8 GB RAM. Our code has been released on GitHub at `https://github.com/Leron33/Graph-matching`.

6.1 Performance Comparison with Synthetic Data

For our experiments on synthetic data, we generate G_1, G_2 and π^* according to the correlated Erdős-Rényi model. We calculate the accuracy rate as the median of the proportion of vertices that are correctly matched, taken over 10 independent simulations.

6.1.1 Performance Comparison between Seeded Algorithms

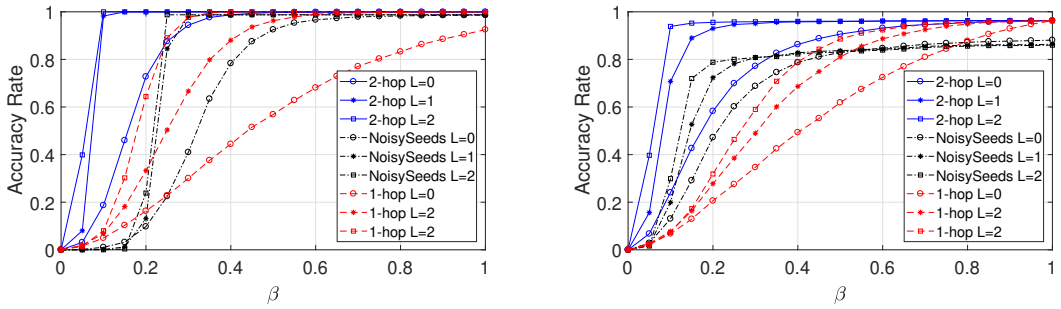
In this section, we compare the graph matching algorithms using partially-correct seeded. In Fig. 6, we fix $n = 10000$ and $s = 0.9$, and plot the accuracy rates for $p = n^{-\frac{3}{4}}$ and $p = n^{-\frac{6}{7}}$. We observe that the 2-hop algorithm significantly outperforms the 1-hop algorithm. For the NoisySeeds algorithm, its performance is sensitive to the threshold value r. The 2-hop algorithm performs either comparably or better than the NoisySeeds algorithm even with the best choice of r. Note that it is *a priori* unclear how to choose the best value of r for the NoisySeeds algorithm, while our 2-hop algorithm does not need to tune any parameters. Computationally, when we match two graphs of size 10000 with $p = n^{-\frac{6}{7}}$ and $\beta = 0.5$, the average running times of the 1-hop algorithm, 2-hop algorithm, and NoisySeeds algorithm are about 52s, 86s and 101s, respectively. Similar to the NoisySeeds algorithm, we can modify GMWM for parallel implementation to make our 2-hop algorithm even more scalable. Please refer to Appendix B for details.



(a) $p = n^{-\frac{3}{4}}$. (b) $p = n^{-\frac{6}{7}}$.

Figure 6: Performance comparison of the 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm with $p = n^{-\frac{3}{4}}$ and $p = n^{-\frac{6}{7}}$. Fix $n = 10000$ and $s = 0.9$.

In Lubars and Srikant (2018), the authors suggest iteratively applying the 1-hop algorithm to further boost its accuracy. Thus, we further iteratively apply the 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm and compare their performance. By using the matching output of the previous iteration as the new partially-correct seeds for the next iteration, we run the three algorithms with a given number of iterations $L = 0, 1, 2$. In Fig. 7, we consider the same setup as in Fig. 6. We fix $n = 10000$ and $s = 0.9$, and plot the accuracy rates for $p = n^{-\frac{3}{4}}$ and $p = n^{-\frac{6}{7}}$. For the NoisySeeds algorithm, we choose the threshold $r = 3$ for $p = n^{-\frac{3}{4}}$ and $r = 2$ for $p = n^{-\frac{6}{7}}$. We observe that iteratively applying these algorithms boost their performance and the 2-hop algorithm still performs the best among the three algorithms when the number of iterations is the same. In particular, when $p = n^{-\frac{6}{7}}$, while the matching accuracy of the 2-hop with multiple iterations gets close to 1, the matching accuracy of NoisySeeds saturates at $0.8 \sim 0.9$. This is because about 10% true pairs have only one common 1-hop neighbor and thus cannot be correctly matched by the NoisySeeds algorithm with $r = 2$.



(a) $p = n^{-\frac{3}{4}}$. (b) $p = n^{-\frac{6}{7}}$.

Figure 7: Performance comparison of the 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm applied iteratively with $p = n^{-\frac{3}{4}}$ and $p = n^{-\frac{6}{7}}$. Fix $n = 10000$ and $s = 0.9$.

6.1.2 Performance Comparison between seeded and seedless algorithms

In this section, we compare the performance of the seeded algorithms (the 1-hop, 2-hop algorithms, and the NoisySeeds algorithm) with that of the seedless algorithms (including degree profile matching algorithm (Ding et al., 2021), quadratic programming relaxation of QAP based on doubly stochastic relaxation (QP), and GRAph Matching by Pairwise eigen-Alignments (GRAMPA) (Fan et al., 2019b)). The results are shown in Fig. 8 as a function of s with $n = 1000$, $p = n^{-3/4}$ and $\beta = 0.5$. Clearly, our 2-hop algorithm outperforms other algorithms, including the seedless ones.

6.1.3 Comparison between GMWM and the Hungarian algorithm

In this section, we compare the Greedy Maximum Weight Matching algorithm with the Hungarian matching algorithm (for solving the linear assignment problem), when they are used as a component of the 1-hop algorithm and the 2-hop algorithm. In Figure 9 be-

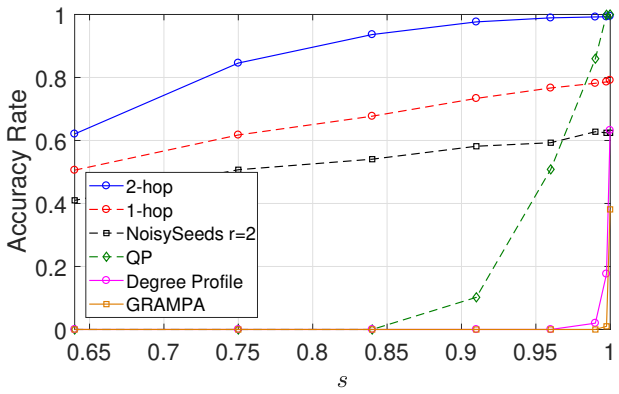


Figure 8: Performance comparison on correlated Erdős-Rényi random graph model with $n = 1000$, $p = n^{-3/4}$ and $\beta = 0.5$.

low, we fix $n = 1000$, $s = 0.9$, and plot the matching accuracy rates for $p = n^{-\frac{3}{4}}$. The simulation results show that the Hungarian algorithm has slightly better matching accuracy. However, the improvement over GMWM is quite small. Further, the running time of the Hungarian algorithm is much larger. Specifically, the average running time of using GMWM algorithm and the Hungarian algorithm in the 1-hop algorithm is about 0.12s and 1.1s, respectively. Thus, the GMWM algorithm saves a lot of running time. In summary, the GMWM algorithm strikes a good balance between time complexity and accuracy.

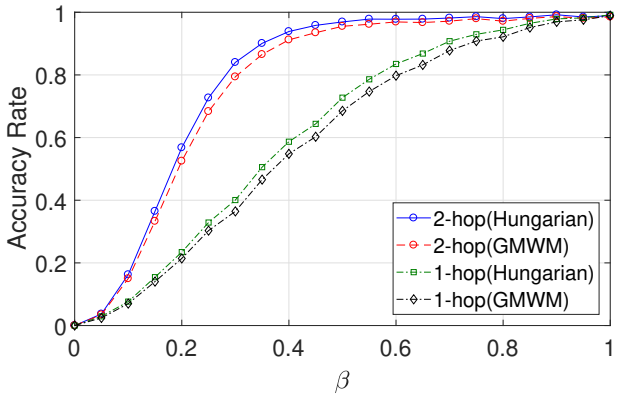


Figure 9: Performance comparison of using GMWM and the Hungarian algorithm in the 1-hop algorithm and the 2-hop algorithm.

6.2 Performance Comparison with Real Data

In this section, we will show that our 2-hop algorithm also performs well on real-world graphs. Further, departing from our simulation with synthetic data where the two corre-

lated graphs have the same number of vertices, we will evaluate the performance of the algorithms when the two correlated graphs have a different number of vertices. In Section 6.2.1, we consider de-anonymizing social networks, which is a popular application of graph matching. In this example, the two correlated graphs are generated by sampling a Facebook friendship network. In Section 6.2.2, we match networks of Autonomous Systems in which the correlated graphs are povided by the real data set. In both Section 6.2.1 and Section 6.2.2, the initial seeds are chosen uniformly randomly. In Section 6.2.3, we match deformable 3D shapes, where not only the correlated graphs are provided by the real data set, but also the initial seeds are generated by a seedless graph match algorithm.

6.2.1 Facebook Friendship Networks

We use a Facebook friendship network of 11621 students and staffs from Standford university provided in Traud et al. (2012) as the parent graph G_0. There are 1136660 edges in G_0. The Facebook social network has an approximate power-law degree distribution with $p(d) \sim d^{-1}$ with average degree about 100. To obtain two correlated subgraphs G_1 and G_2 of different sizes, we independently sample each edge of G_0 twice with probability $s = 0.9$ and sample each vertex of G_0 twice with probability $\alpha = 0.8$. Then, we relabel the vertices in G_2 according to a random permutation $\pi^* : [n_2] \to [n_2]$, where n_2 is the number of nodes in G_2. Let m denote the number of common vertices that appear in both G_1 and G_2. The initial seed mapping is constructed by uniformly and randomly choosing a mapping $\pi : [m] \to [m]$ between the common vertices of the two subgraphs such that β fraction of vertices are correctly matched, i.e., $\pi(u) = \pi^*(u)$ for exactly βm common vertices. We treat G_1 as the public network and G_2 as the private network, and the goal is to de-anonymize the node identities in G_2 by matching G_1 and G_2. We show the performance of the 1-hop algorithm, 2-hop algorithm, and NoisySeeds algorithm in Fig. 10. We choose the threshold $r = 5, 10, 15$ for the NoisySeeds algorithm to search for the best value of r. We observe that our proposed 2-hop algorithm significantly outperforms the 1-hop algorithm and NoisySeeds algorithm. Note that the matching accuracy is saturated at around 80%, because there are about 15% common vertices that are isolated in the intersection graph $G_1^{\pi^*} \wedge G_2$ and thus can not be correctly matched. Due to the power-law degree variation, the number of witnesses for some fake pairs could be larger than the threshold r. Thus, even the NoisySeeds algorithm with the best value of r does not perform well.

6.2.2 Autonomous Systems Networks

Following Fan et al. (2020), we use the Autonomous Systems (AS) data set from Leskovec and Krevl (2014) to test the graph matching performance on real graphs. The data set consists of 9 graphs of Autonomous Systems peering information inferred from Oregon route-views between March 31, 2001, and May 26, 2001. Since some vertices and edges are changed over time, these nine graphs can be viewed as correlated versions of each other. The number of vertices of the 9 graphs ranges from 10,670 to 11,174 and the number of edges from 22,002 to 23,409. The Autonomous Systems networks have an approximate power-law degree distribution with $p(d) \sim d^{-2}$ with average degree about 2. We apply the 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm (with the best-performing threshold $r = 2$) to match each graph to that on March 31, with vertices randomly permuted. To

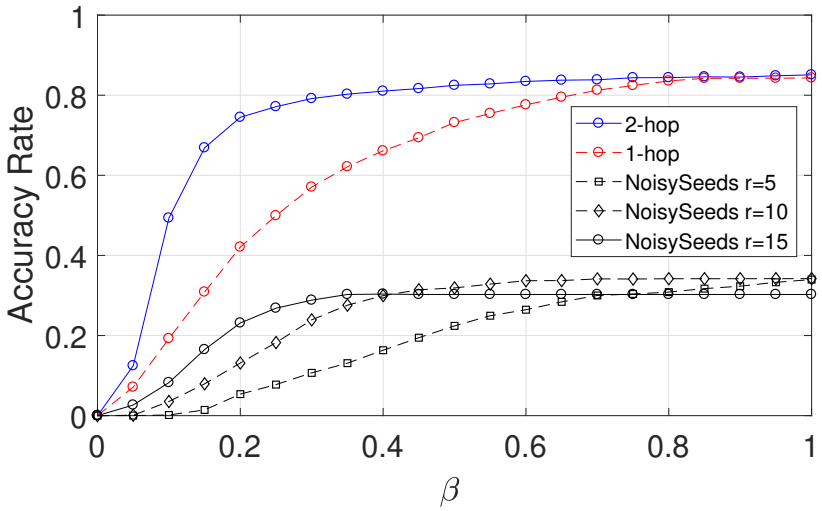


Figure 10: Performance comparison of 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm applied to the Facebook networks.

obtain the initial seed mapping, we uniformly and randomly choose the mapping between the common vertices among the two given graphs such that β fraction of vertices are correctly matched.

The performance comparison of the three algorithms is plotted in Fig. 11 for $\beta = 0.3, 0.6, 0.9$. We observe that our proposed 2-hop algorithm significantly outperforms the 1-hop algorithm and NoisySeeds Algorithm. The NoisySeeds algorithm does not perform well due to its thresholding component: There exists high degree variation in these real graphs and thus a significant fraction of true pairs have only 1 witness, which falls below even the smallest threshold $r = 2$. Note that the accuracy rates for all algorithms decay in time because over time the graphs become less correlated with the initial one on March 31. Computationally, when we match two real graphs with $\beta = 0.6$, the average running time of the 1-hop algorithm, 2-hop algorithm, and NoisySeeds algorithm is about 46s, 73s and 91s, respectively.

6.2.3 Computer Vision Data Set

In this experiment, we use the output of seedless graph matching algorithms as partially correct seeds, and test the performance of 1-hop, 2-hop, and NoisySeeds in correcting the initial matching errors. We focus on the application of deformable shape matching. Matching 3D deformable shapes is a fundamental and ubiquitous problem in computer vision with numerous applications such as object recognition, and has been extensively studied for decades (see Van Kaick et al. (2011) and Sahillioğlu (2020) for surveys). At a high-level, each 3D shape is represented as a mesh graph. For two 3D shapes corresponding to the same object but with different poses, their mesh graphs are approximately isomorphic. However,

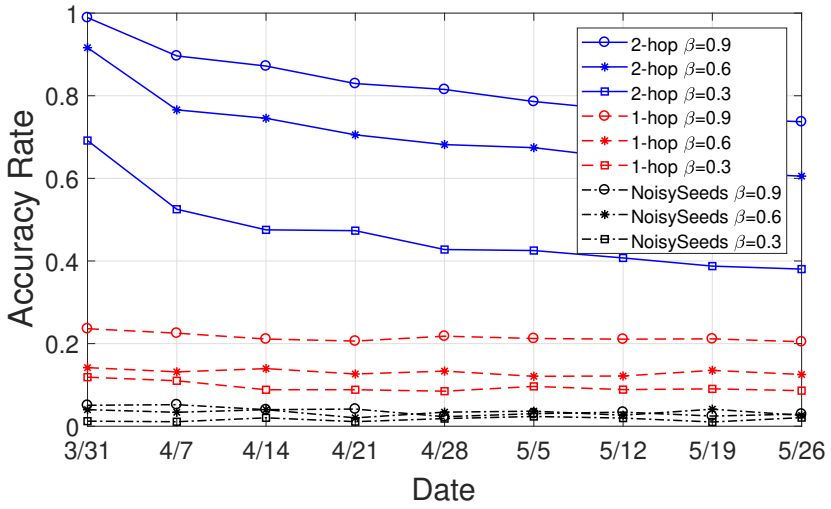


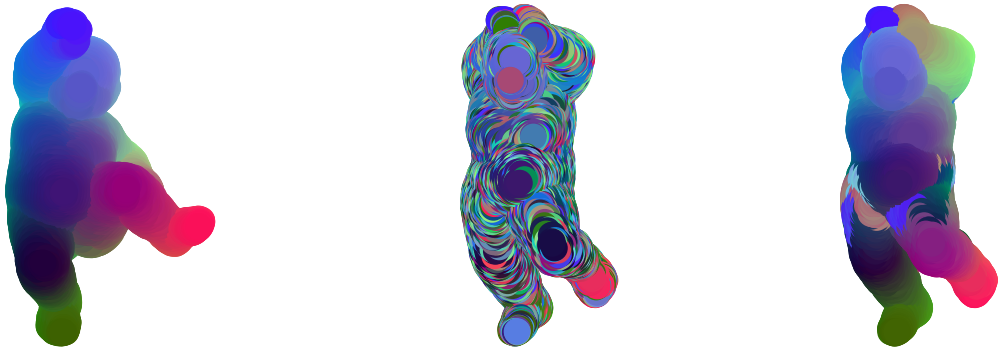
Figure 11: Performance comparison of 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm applied to the Autonomous Systems graphs.

the exact vertex mapping is not knowm. Thus, the goal of deformable shape matching is to retrieve the correct vertex correspondence by matching the two mesh graphs.

The previous work Fan et al. (2020) applied the 1-hop algorithm iteratively to boost the matching accuracy of their seedless graph matching algorithm, GRAMPA (GRAph Matching by Pairwise eigen-Alignments). Their experiment is carried on the SHREC'16 data set in Lähner et al. (2016). The SHREC'16 data set provides 25 deformable 3D shapes (15 for training and 10 for testing) undergoing different topological changes. At the lower resolution, each shape is represented by a triangulated mesh graph consisting of around 8K-11K vertices with 3D coordinates and 17K-22K triangular faces, with vertex degrees highly concentrated on 6. It is demonstrated in Fan et al. (2020) that the GRAMPA followed by the iterative 1-hop algorithm achieves much higher matching accuracy compared to the existing methods tested in Lähner et al. (2016).

We also use the SHRED'16 data set in our experiment. When we match each pair of test shapes, we first apply the GRAMPA algorithm, and then repeatedly use the 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm with 100 iterations to boost the matching accuracy of the output of the GRAMPA algorithm. Fig. 12 provides a visualization of our results, where the matched vertices are colored with the same color. We can see that the 2-hop algorithm corrects most matching errors of the GRAMPA algorithm.

We follow the Princeton benchmark protocol in Kim et al. (2011) to evaluate the matching quality. Assume that a vertex-pair $(i, j) \in \mathcal{M} \times \mathcal{N}$ is matched between shapes $\mathcal{M}$ and $\mathcal{N}$, while the ground-truth correspondence is (i, j^*). Then the normalized geodesic error of this correspondence at vertex i is defined as $\varepsilon(i) = \frac{d_{\mathcal{N}}(j,j^*)}{\sqrt{\text{area}(\mathcal{N})}}$, where $d_{\mathcal{N}}$ denotes the geodesic distance on $\mathcal{N}$ and area$(\mathcal{N})$ is the total surface area of $\mathcal{N}$. Finally, we plot the cumulative



(a) (b) (c)

Figure 12: Visualization of the matching results. Fig. 12(a) is the mesh graph of a 3D shape randomly chosen from the SHREC'16 data set, whose vertices are colored in a gradient. Fig. 12(b) and Fig. 12(c) are the same mesh graph with a different pose that needs to be matched with Fig. 12(a), where vertices in Fig. 12(b) and Fig. 12(c) are labeled with the same color as that of the vertices in Fig. 12(a) matched by the corresponding graph matching algorithms. Fig. 12(b) shows the matching result of the GRAMPA algorithm. Fig. 12(c) shows the matching result when we use the output of the GRAMPA algorithm as the initial seeds and apply the 2-hop algorithm iteratively. We can observe that the 2-hop algorithm corrects most matching errors of the GRAMPA algorithm.

distribution function of $\{\varepsilon(i)\}_{i=1}^{n}$ in Fig. 13, where $\mathrm{cdf}(\epsilon)$ is the fraction of vertices i such that $\varepsilon(i) \leq \epsilon$. In particular, $\mathrm{cdf}(0)$ is the fraction of correctly matched vertices in shape $\mathcal{M}$.

In Fig. 13, we observe that all three algorithms improve the initial matching accuracy of the GRAMPA algorithm, but the performance improvement of our 2-hop algorithm is most substantial. In particular, our 2-hop algorithm increases the fraction of correctly matched vertices to more than 80%, while the 1-hop algorithm and NoisySeeds (with the best choice threshold $r = 2$) only correctly match around 60% and 30% of vertex-pairs, respectively.

7. Conclusion

In this work, we tackle the graph matching problem with partially-correct seeds. Under the correlated Erdős-Rényi model, we first present a sharper characterization of the condition for the 1-hop algorithm to perfectly recover the vertex matching for dense graphs, which requires many fewer correct seeds than the prior art when graphs are dense. Then, for sparse graphs, by exploiting 2-hop neighbourhoods, we propose an efficient 2-hop algorithm that perfectly recovers the true vertex correspondence with even fewer correct seeds than the 1-hop algorithm in sparse graphs. Our performance guarantees for the 1-hop and 2-hop algorithm combined together achieve the best-known results across the entire range of

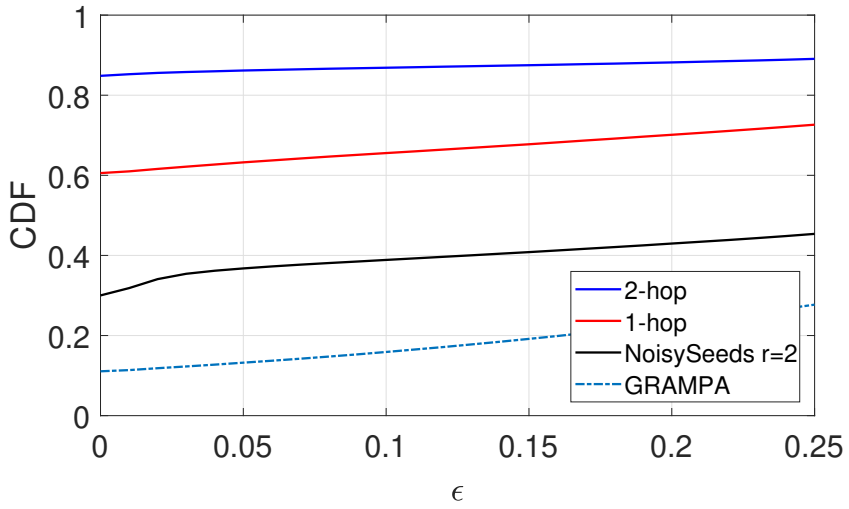


Figure 13: Performance comparison of 1-hop algorithm, 2-hop algorithm and NoisySeeds algorithm applied to the SHREC'16 data set, with initial noisy seeds generated by the GRAMPA algorithm. The higher the curve, the better the algorithm performance.

graph sparsity and significantly improve the state-of-the-art. Moreover, our results precisely characterize the graph sparsity at which the 2-hop algorithm starts to outperform 1-hop. This reveals an interesting and delicate trade-off between the *quantity* and the *quality* of witnesses: while the 2-hop algorithm exploits more seeds as witnesses than 1-hop, the 2-hop witnesses are less accurate than the 1-hop counterparts in distinguishing true pairs from fake pairs when graphs are dense.

Experimental results validate our theoretical analysis, demonstrating that our 2-hop algorithm continues to perform well in real graphs with power-law degree variations and different number of nodes. There are many exciting future directions such as analyzing the performance of j-hop algorithms for $j \geq 3$, investigating partial recovery when $nps^2 - \log n = O(1)$, and studying graph matching under other random graph models beyond Erdős-Rényi random graphs.

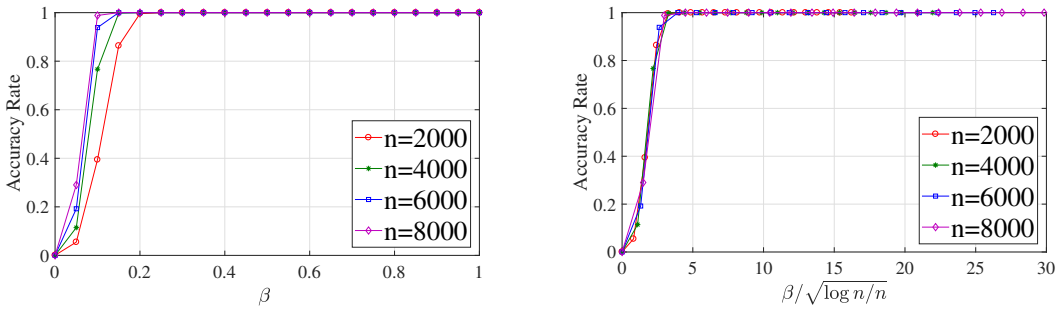
Acknowledgments

L. Yu and J. Xu are supported by the NSF Grant IIS-1932630.

Appendix A. Numerical Experiments to Verify The Scalings

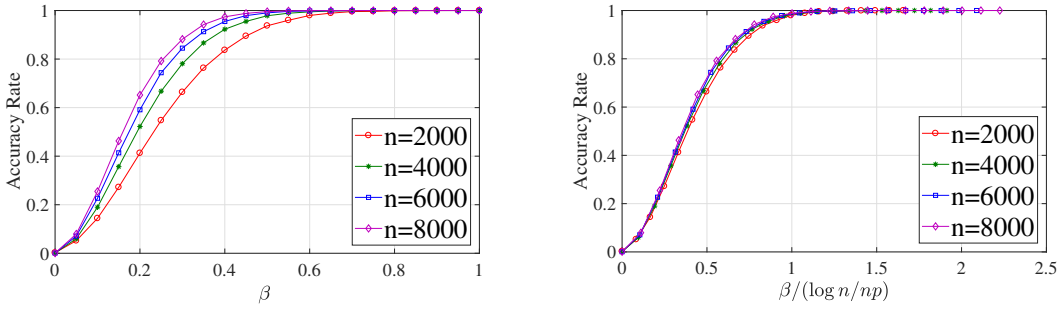
In this section, we conduct numerical studies to verify the scaling results given in Theorem 1 and Theorem 2. We observe that conditions (3) and (4) are not only sufficient, but also close to necessary (differing from the necessary conditions by a constant factor) for the 1-hop and 2-hop algorithms to succeed, respectively. Throughout, we generate G_1, G_2 and π^* according to the correlated Erdős-Rényi model with fixed sampling probability $s = 0.8$, and vary the number of vertices from 2000 to 8000.

We first simulate the performance of the 1-hop algorithm for $p = n^{-\frac{1}{3}}$ and $p = n^{-\frac{2}{3}}$. The results are presented in Fig. 14(a) and Fig. 15(a) as a function of β. Theorem 1 predicts that the 1-hop algorithm succeeds in exact recovery when $\beta \gtrsim \sqrt{\log n/n}$ for $p = n^{-\frac{1}{3}}$ and when $\beta \gtrsim \frac{\log n}{np}$ for $p = n^{-\frac{2}{3}}$. Thus, we rescale the x-axis in Fig. 14(b) and Fig. 15(b) as $\beta/\sqrt{\frac{\log n}{n}}$ and $\beta/\left(\frac{\log n}{np}\right)$, respectively. We see that after rescaling the curves for different n align well with each other, suggesting that condition (3) is both sufficient and close to necessary for the 1-hop algorithm to succeed.



(a) x-axis is β. (b) x-axis is $\beta/\sqrt{\log n/n}$.

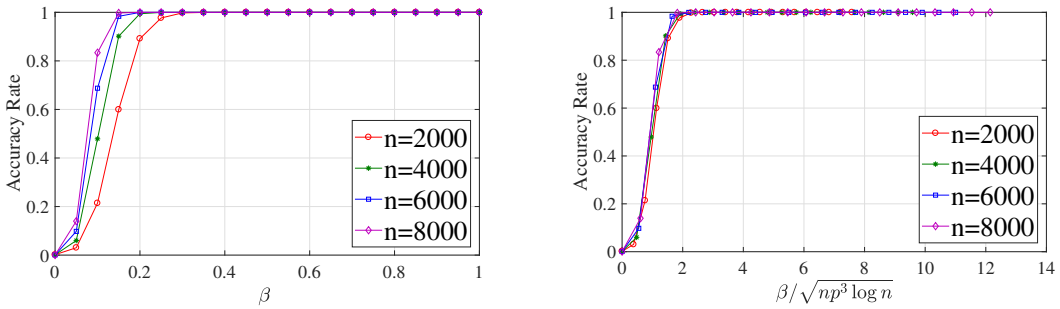
Figure 14: The 1-hop algorithm with varying n and $p = n^{-\frac{1}{3}}$. Fix $s = 0.8$.



(a) x-axis is β. (b) x-axis is $\beta/(\log n/np)$.

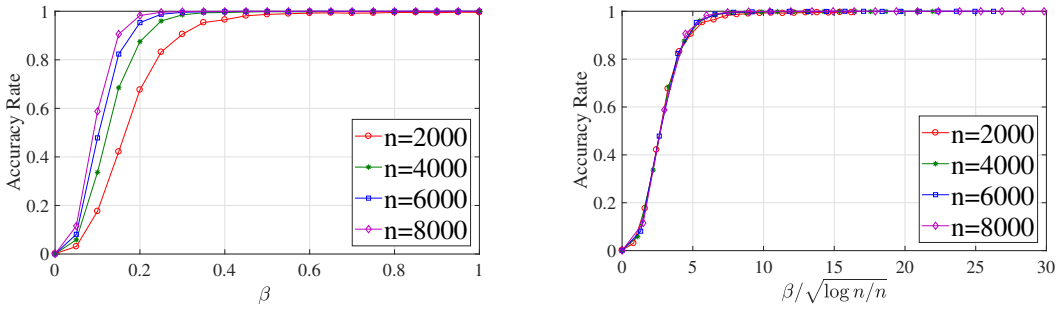
Figure 15: The 1-hop algorithm with varying n and $p = n^{-\frac{2}{3}}$. Fix $s = 0.8$.

Next, we simulate the performance of the 2-hop algorithm for $p = n^{-\frac{3}{5}}$, $p = n^{-\frac{17}{24}}$, and $p = n^{-\frac{4}{5}}$. The results are presented in Fig. 16(a), Fig. 17(a) and Fig. 18(a). Since Theorem 2 predicts that the 2-hop algorithm succeeds in exact recovery with high probability when $\beta \gtrsim \max\left\{\sqrt{np^3 \log n}, \sqrt{\frac{\log n}{n}}, \frac{\log n}{n^2 p^2}\right\}$, we rescale the x-axis in Fig. 16(b), Fig. 17(b) and Fig. 18(b) as $\beta/\sqrt{np^3 \log n}$ for $p = n^{-\frac{3}{5}}$, $\beta/\sqrt{\frac{\log n}{n}}$ for $p = n^{-\frac{17}{24}}$ and $\beta/\left(\frac{\log n}{n^2 p^2}\right)$ for $p = n^{-\frac{4}{5}}$. As we can see in Fig. 16(b), Fig. 17(b) and Fig. 18(b), the curves for different n align well with each other, suggesting that condition (4) is both sufficient and close to necessary for the 2-hop algorithm to succeed.



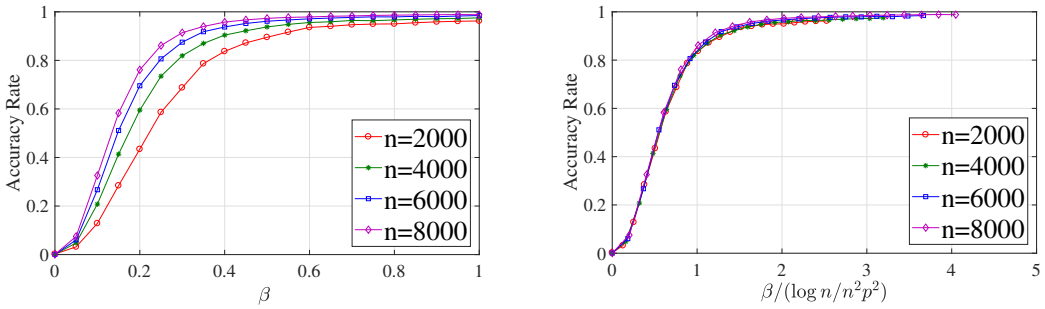
(a) x-axis is β. (b) x-axis is $\beta/\sqrt{np^3 \log n}$.

Figure 16: The 2-hop algorithm with varying n and $p = n^{-\frac{3}{5}}$. Fix $s = 0.8$.



(a) x-axis is β. (b) x-axis is $\beta/\sqrt{\log n/n}$.

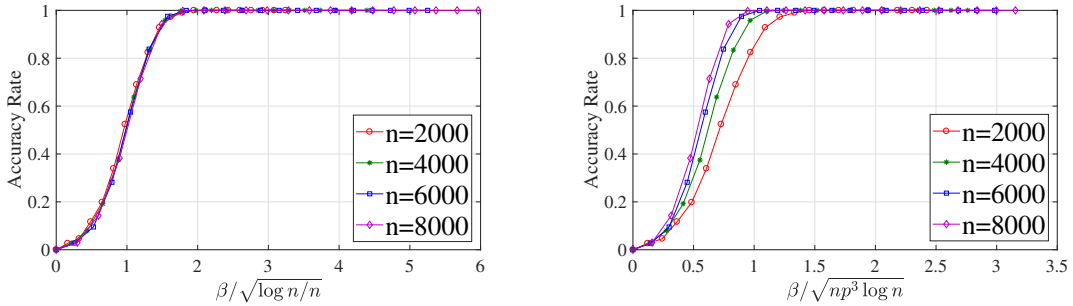
Figure 17: The 2-hop algorithm with varying n and $p = n^{-\frac{17}{24}}$. Fix $s = 0.8$.



(a) x-axis is β. (b) x-axis is $\beta/(\log n/n^2p^2)$.

Figure 18: The 2-hop algorithm with varying n and $p = n^{-\frac{4}{5}}$. Fix $s = 0.8$.

In addition, if $s = 1$ and $p = n^{-\frac{1}{2}}$, we show in Fig. 19 that the curves for different n align well when we rescale the x-axis as $\beta/\sqrt{\frac{\log n}{n}}$, but they do not align well with each other when the x-axis is rescaled as $\beta/\sqrt{np^3 \log n}$. This result agrees with Theorem 2, demonstrating that condition (25) derived from the old criteria (23) is not tight.



(a) x-axis is $\beta/\sqrt{\log n/n}$. (b) x-axis is $\beta/\sqrt{np^3 \log n}$.

Figure 19: The 2-hop algorithm with varying n and $p = n^{-\frac{1}{2}}$. Fix $s = 1$.

Appendix B. The scalability of our algorithm and feasible parallel implementation

We may further improve the time complexity of our 2-hop algorithm by exploiting graph sparsity and parallel computing. Recall that the theoretical worst-case computational complexity of our algorithm is $O(n^\omega + n^2 \log n)$ for $2 \le \omega \le 2.373$, where n^ω denotes the complexity of $n \times n$ matrix multiplication. For sparse graphs, the computational complexity of our 2-hop algorithm is comparable to that of the NoisySeeds algorithm in Kazemi et al. (2015) and the 1-hop algorithm in Lubars and Srikant (2018). To see this, note that there are only two differences in the execution of our 2-hop algorithm: (i) To compute the number of 2-hop witnesses for all vertex-pairs, for every seed $(w, \pi(w))$, our algorithm needs to compute the set of 2-hop neighbors of w (resp. $\pi(w)$) in G_1 (resp. G_2), which takes $O\left(c^4\right)$

steps, where c denotes the average degree. Thus in total it takes $O\left(nc^4\right)$ steps. Hence, for sparse graphs with small average degree c, finding 2-hop witnesses only increases complexity slightly compared to finding 1-hop witnesses; (ii) We use greedy max weight matching (GMWM) rather than simple thresholding in Kazemi et al. (2015). As the time complexity of GMWM is $O(n^2 \log n)$ and the thresholding procedure needs to go through all the n^2 vertex-pairs, GMWM only incurs an additional $\log n$ factor to time complexity. Thus, the computational complexity of our 2-hop algorithm is comparable to others.

For very large graphs, one may want to run these algorithms parallelly. Our 2-hop algorithm can be executed in parallel as follows. First, it is easy to turn (i) into parallel implementation. Second, if the complexity of (ii) is an issue, we can instead run the following modification: For each vertex u in G_1, matches it to v in G_2 that has the largest number of 2-hop witnesses; Output failure if there is any inconsistency in the final matching. This procedure can then be executed across all nodes in G_1 (or G_2) in parallel. This parallelizable version of the 2-hop algorithm can still provide perfect recovery if criteria (23) holds. This criteria is satisfied with high probability under condition (25), as discussed in Remark 5.2.3. Hence the parallelizable version of the 2-hop algorithm can achieve perfect recovery under condition (25). Thus, we believe our 2-hop algorithm can scale to very large graphs with strong matching performance.

Appendix C. Preliminary Results

We first present some useful concentration inequalities for the sum of independent random variables.

Theorem 13 *Chernoff Bound (Dubhashi and Panconesi (2009)): Let $X = \sum_{i\in[n]} X_i$, where X_i's are independent random variables taking values in $\{0,1\}$. Then, for $\delta \in (0,1)$,*

$$\mathbb{P}\left\{X \le (1-\delta)\mathbb{E}\left[X\right]\right\} \le \exp\left(-\frac{\delta^2}{2}\mathbb{E}\left[X\right]\right),\ \mathbb{P}\left\{X \ge (1+\delta)\mathbb{E}\left[X\right]\right\} \le \exp\left(-\frac{\delta^2}{3}\mathbb{E}\left[X\right]\right).$$

As a corollary of Theorem 13, we obtain the following lemma, which will be useful for the proofs of Lemma 6 and Lemma 11.

Lemma 14 *Let X denote a random variable such that $X \sim \text{Binom}(n-1,\alpha)$. If $\alpha \in \left[ps^2, 1\right)$ and $nps^2 \ge 128\log n$, then*

$$\mathbb{P}\left\{X \le (1-\epsilon)(n-1)\alpha\right\} \le n^{-6},\ \mathbb{P}\left\{X \ge (1+\epsilon)(n-1)\alpha\right\} \le n^{-4},$$

where ϵ is given in (12), i.e., $\epsilon = \sqrt{\frac{12\log n}{(n-1)ps^2}} \le \frac{1}{3}$.

Proof Since $X \sim$ Binom $(n-1,\alpha)$ and $\epsilon = \sqrt{\frac{12\log n}{(n-1)ps^2}} < \frac{1}{3}$, applying Chernoff bound in Theorem 13 and using $\alpha \ge ps^2$ yields

$$\mathbb{P}\left\{X \le (1-\epsilon)(n-1)\alpha\right\} \le \exp\left(-\frac{\epsilon^2(n-1)\alpha}{2}\right) \le n^{-6},$$
$$\mathbb{P}\left\{X \le (1+\epsilon)(n-1)\alpha\right\} \le \exp\left(-\frac{\epsilon^2(n-1)\alpha}{3}\right) \le n^{-4}.$$

■

Theorem 15 *Bernstein's Inequality (Dubhashi and Panconesi (2009)): Let $X = \sum_{i\in[n]} X_i$, where X_i's are independent random variables such that $|X_i| \leq K$ almost surely. Then, for $t > 0$, we have*

$$\mathbb{P}\left\{X \geq \mathbb{E}\left[X\right] + t\right\} \leq \exp\left(-\frac{t^2}{2(\sigma^2 + Kt/3)}\right),$$

where $\sigma^2 = \sum_{i\in[n]} \mathsf{var}(X_i)$ is the variance of X. It follows then for $\gamma > 0$, we have

$$\mathbb{P}\left\{X \geq \mathbb{E}\left[X\right] + \sqrt{2\sigma^2\gamma} + \frac{2K\gamma}{3}\right\} \leq \exp(-\gamma).$$

Similarly, by considering $-X$, it follows that

$$\mathbb{P}\left\{X \leq \mathbb{E}\left[X\right] - \sqrt{2\sigma^2\gamma} - \frac{2K\gamma}{3}\right\} \leq \exp(-\gamma).$$

Corollary 16 *Let X denote a random variable such that $X \sim \mathrm{Binom}(n, \alpha)$. If $n \in [n_{\min}, n_{\max}]$, then for $\gamma > 0$,*

$$\mathbb{P}\left\{X \leq n_{\min}\alpha - \sqrt{2n_{\max}\alpha\gamma} - \frac{2\gamma}{3}\right\} \leq \exp(-\gamma), \tag{31}$$

$$\mathbb{P}\left\{X \geq n_{\max}\alpha + \sqrt{2n_{\max}\alpha\gamma} + \frac{2\gamma}{3}\right\} \leq \exp(-\gamma). \tag{32}$$

Moreover,

$$\mathbb{P}\left\{X \geq 2n_{\max}\alpha + \frac{4\gamma}{3}\right\} \leq \exp(-\gamma) \tag{33}$$

Proof The proof of (31) and (32) follows by invoking Theorem 15 with $\sigma^2 = n\alpha(1-\alpha)$ and $K = 1$ and using the assumption that $n \in [n_{\min}, n_{\max}]$. In view of $2\sqrt{ab} \leq a + b$, (33) follows from (32). ■

Next, we present a concentration inequality for the sum of dependent random variables. To this end, we first introduce the notion of dependency graph.

Definition 17 *Given random variables $\{X_i\}_{i\in[n]}$, the dependency graph is a graph Γ with vertex set $[n]$ such that if $i \in [n]$ is not connected by an edge to any vertex in $\mathcal{J} \subset [n]$, then X_i is independent of $\{X_j\}_{j\in\mathcal{J}}$.*

Theorem 18 *(Janson (2004)) Let $X = \sum_{i\in[n]} X_i$, where X_i's are random variables such that $X_i - \mathbb{E}\left[X_i\right] \leq K$ for some $K > 0$. Let Γ denote a dependency graph for $\{X_i\}$ and $\Delta(\Gamma)$ denote the maximum degree of Γ. Let $\sigma^2 = \sum_{i\in[n]} \mathsf{var}(X_i)$. Then, for $t \geq 0$,*

$$\mathbb{P}\left\{X \geq \mathbb{E}\left[X\right] + t\right\} \leq \exp\left(-\frac{8t^2}{25\Delta_1(\Gamma)(\sigma^2 + Kt/3)}\right),$$

where $\Delta_1(\Gamma) = \Delta(\Gamma) + 1$. *It follows then for* $\gamma > 0$, *we have*

$$\mathbb{P}\left\{X \geq \mathbb{E}[X] + \sqrt{\frac{25\Delta_1(\Gamma)}{8}\sigma^2\gamma} + \frac{25\Delta_1(\Gamma)K\gamma}{24}\right\} \leq \exp(-\gamma).$$

If the assumption $X_i - \mathbb{E}[X_i] \leq K$ *is reversed to* $X_i - \mathbb{E}[X_i] \geq -K$, *then by considering* $-X$, *it follows that*

$$\mathbb{P}\left\{X \leq \mathbb{E}[X] - \sqrt{\frac{25\Delta_1(\Gamma)}{8}\sigma^2\gamma} - \frac{25\Delta_1(\Gamma)K\gamma}{24}\right\} \leq \exp(-\gamma).$$

Finally, we will repeatedly use the following simple inequality.

Theorem 19 *For* $r \geq 0$, *every real number* $x \in (0,1)$ *and* $rx \leq 1$, *it holds that*

$$r\log(1-x) \leq \log\left(1 - \frac{rx}{2}\right).$$

Proof Set $f(x) = r\log(1-x) - \log\left(1-\frac{rx}{2}\right)$. Then $f(0) = 0$ and $f'(x) = \frac{r(rx-x-1)}{(2-rx)(1-x)} \leq 0$. Thus $f(x) \leq 0$, completing the proof. ■

Appendix D. Postponed Proofs for Theorem 1

D.1 Proof of Lemma 4

Recall that A_1 and B_1 are the adjacency matrix for G_1 and G_2, respectively. By the definition of 1-hop witness, we have

$$W_1(u,v) = \sum_{i \in [n] \setminus \{u, \pi^{-1}(v)\}} A_1(u,i)B_1(v,\pi(i)). \tag{34}$$

Let $Z_i \triangleq A_1(u,i)B_1(v,\pi(i))$. Note that Z_v is dependent on $Z_{\pi^{-1}(u)}$. Thus, we exclude these two seeds and consider the remaining seeds. For all $i \in [n] \setminus \{u, v, \pi^{-1}(u), \pi^{-1}(v)\}$, $Z_i \overset{\text{i.i.d.}}{\sim} \text{Bern}(p^2s^2)$. It follows that

$$\mathbb{P}\left\{\sum_{i \in [n] \setminus \{u,v,\pi^{-1}(u),\pi^{-1}(v)\}} Z_i \geq \psi_{\max} - 2\right\} \leq \mathbb{P}\left\{\text{Binom}(n, p^2s^2) \geq \psi_{\max} - 2\right\} \leq n^{-\frac{7}{2}}, \tag{35}$$

where that last inequality follows from Bernstein's inequality given in Theorem 15 with $\gamma = \frac{7}{2}\log n$ and $K = 1$.

Finally, adding back Z_v and $Z_{\pi^{-1}(u)}$ yields the desired conclusion (6).

D.2 Proof of Theorem 1

Since the bound of the number of 1-hop witnesses is provided by Lemma 4 and Lemma 5, it remains to verify $x_{\min} + y_{\min} - \psi_{\max} \geq 0$ under the condition of Theorem 1. Note that

$$\begin{aligned}&x_{\min} + y_{\min} - \psi_{\max}\\ =&n\beta p(1-p)s^2 - \sqrt{5n\beta ps^2 \log n} - (5+\sqrt{7})\sqrt{np^2s^2\log n} - ps^2 - 2p^2s^2 - \frac{37}{3}\log n - 2.\end{aligned}$$

First, by assumption that $\beta \geq \frac{45\log n}{np(1-p)^2s^2}$, we have

$$\frac{1}{3}n\beta p(1-p)s^2 \geq \sqrt{\frac{45\log n}{np(1-p)^2s^2}} \cdot \frac{1}{3}n\sqrt{\beta}p(1-p)s^2 = \sqrt{5n\beta ps^2\log n}. \tag{36}$$

Second, by assumption that $\beta \geq 30\sqrt{\frac{\log n}{n(1-p)^2s^2}}$, we have

$$\frac{1}{3}n\beta p(1-p)s^2 \geq 30\sqrt{\frac{\log n}{n(1-p)^2s^2}} \cdot \frac{1}{3}np(1-p)s^2 \geq (5+\sqrt{7})\sqrt{np^2s^2\log n}. \tag{37}$$

Third, by assumption that $\beta \geq \frac{45\log n}{np(1-p)^2s^2}$ and n is sufficiently large, we have

$$\frac{1}{3}n\beta p(1-p)s^2 \geq \frac{45\log n}{np(1-p)^2s^2} \cdot \frac{1}{3}np(1-p)s^2 \geq \frac{37}{3}\log n + 2 + ps^2 + 2p^2s^2. \tag{38}$$

Combining (36)-(38), we have $x_{\min} + y_{\min} - \psi_{\max} \geq 0$. Thus,

$$\begin{aligned}&\mathbb{P}\left\{\min_{u\in[n]} W_1(u,u) > \max_{u,v\in[n]:u\neq} W_1(u,v)\right\}\\ \geq&1 - \mathbb{P}\left\{\bigcup_{u\in[n]}\{W_1(u,u) \leq x_{\min} + y_{\min}\}\right\} - \mathbb{P}\left\{\bigcup_{u,v\in[n]:u\neq v}\{W_1(u,v) \geq \psi_{\max}\}\right\}\\ \geq&1 - n^{-\frac{4}{3}} - n^{-\frac{3}{2}} \geq 1 - n^{-1},\end{aligned}$$

where the second inequality holds by combining Lemma 4 and Lemma 5 with the union bound. Thus, GMWM outputs $\widetilde{\pi}$ with $\mathbb{P}\{\widetilde{\pi} = \pi^*\} \geq 1 - n^{-1}$ under the 1-hop algorithm.

Appendix E. Postponed Proofs for Theorem 2

E.1 Proof of Lemma 6

By definition, we have $a_u \sim \text{Binom}(n-1, ps)$. It follows from Lemma 14 that

$$\mathbb{P}\{a_u \leq (1-\epsilon)(n-1)ps\} \leq n^{-6},\ \mathbb{P}\{a_u \geq (1+\epsilon)(n-1)ps\} \leq n^{-4}. \tag{39}$$

The same lower and upper bounds hold for b_u analogously.

Note that $c_{uu} \sim \text{Binom}(n-1, ps^2)$. Applying Lemma 14 yields that

$$\mathbb{P}\left\{c_{uu} \leq (1-\epsilon)(n-1)ps^2\right\} \leq n^{-6},\ \mathbb{P}\left\{c_{uu} \geq (1+\epsilon)(n-1)ps^2\right\} \leq n^{-4}. \tag{40}$$

Also, for fake pairs $u \neq v$, $c_{uv} \sim \text{Binom}(n-2, p^2s^2)$. Therefore, applying Bernstein's inequality given in Theorem 15 with $\gamma = \frac{7}{2}\log n$ and $K = 1$, we can get

$$\mathbb{P}\left\{c_{uv} \geq \psi_{\max}\right\} \leq \mathbb{P}\left\{\text{Binom}(n-2, p^2s^2) \geq np^2s^2 + \sqrt{7np^2s^2\log n} + \frac{7}{3}\log n\right\} \leq n^{-\frac{7}{2}}. \tag{41}$$

According to Lemma 4, we can get

$$\mathbb{P}\left\{W_1(v,u) \geq \psi_{\max}\right\} \leq n^{-\frac{7}{2}}. \tag{42}$$

Taking the union bound over (39)–(42) yields the desired conclusion (13).

E.2 Proof of Lemma 7

Fixing any two vertices $u \neq v$, we condition on Q_{uv} such that the event R_{uv} holds. Note that

$$W_2(u,u) = \sum_{j=1}^{n} A_2(u,j)B_2\left(u, \pi(j)\right),$$

where A_2 and B_2 are the 2-hop adjacency matrix of G_1 and G_2, respectively. Note that for all $j \in N^{G_1}(u) \cup \{u\}$, $A_2(u,j) = 0$ by definition. Similarly, for all $\pi(j) \in N^{G_2}(u) \cup \{u\}$, $B_2\left(u, \pi(j)\right) = 0$. Thus, we define

$$J_u = N^{G_1}(u) \cup \{u\} \cup \pi^{-1}\left(N^{G_2}(u)\right) \cup \pi^{-1}(u)$$

and exclude the seeds in J_u. Furthermore, note that we have conditioned on the 1-hop neighborhoods of v in G_1 and G_2. In either G_1 or G_2, if u and v are connected, then a 1-hop neighbor of v may automatically become the 2-hop neighbor of u. Hence, if j is connected to v in G_1 or $\pi(j)$ is connected to v in G_2, then conditioning on Q_{uv} can change the probability that $A_2(u,j)B_2\left(u, \pi(j)\right) = 1$. To circumvent this issue, we further exclude the set J_v of seeds and get that

$$\begin{aligned} W_2(u,u) &\geq \sum_{j \in [n] \setminus (J_u \cup J_v)} A_2(u,j)B_2\left(u, \pi(j)\right) \\ &= \sum_{j \in F \setminus (J_u \cup J_v)} A_2(u,j)B_2\left(u, j\right) + \sum_{j \in [n] \setminus (F \cup J_u \cup J_v)} A_2(u,j)B_2\left(u, \pi(j)\right), \end{aligned} \tag{43}$$

where $F = \{j : \pi(j) = j\}$ corresponds to the set of correct seeds with $|F| = n\beta$. Since the event R_{uv} holds, it follows that $|J_u \cup J_v| \leq 4(1+\epsilon)(n-1)ps + 4 \leq 6nps$, where the last inequality holds due to $\epsilon \leq 1/3$ and $nps \geq 6$. Thus, $n_{\mathrm{R}} \triangleq |F \setminus (J_u \cup J_v)| \geq n(\beta - 6ps)$.

We first count the contribution to $W_2(u,u)$ by correct seeds. For each correct seed $j \in F \setminus (J_u \cup J_v)$, define an indicator variable $\chi_j = \mathbf{1}_{\{\exists i \in C(u,u) \setminus \{v\} : j \in C(i,i)\}}$. In other words, $\chi_j = 1$ if j is connected to some "common" 1-hop neighbor of true pair (u,u) in both G_1 and G_2, and $\chi_j = 0$ otherwise. By definition $A_2(u,j)B_2\left(u,j\right) \geq \chi_j$. Moreover,

$$\mathbb{P}\left\{\chi_j = 1 \mid Q_{uv}\right\} = 1 - \mathbb{P}\left\{\bigcap_{i \in C(u,u) \setminus \{v\}} \{j \notin C(i,i)\} \mid Q_{uv}\right\}$$

$$\begin{aligned}
&\overset{(a)}{=}1-\prod_{i\in C(u,u)\setminus\{v\}}\mathbb{P}\left\{j\notin C(i,i)\mid Q_{uv}\right\}\\
&\overset{(b)}{=}1-\left(1-ps^2\right)^{|C(u,u)\setminus\{v\}|}\\
&\overset{(c)}{\geq}\frac{1}{2}\left(c_{uu}-1\right)ps^2\overset{(d)}{\geq}\frac{7}{24}np^2s^4,
\end{aligned}$$

where (a) holds because $\{j\notin C(i,i)\}=\{A_1(i,j)=0\}\cup\{B_1(i,j)=0\}$, which are independent across different vertices i conditional on Q_{uv}; (b) holds due to $\mathbb{P}\{j\in C(i,i)\}=\mathbb{P}\{A_1(i,j)=B_1(i,j)=1\}=ps^2$; (c) follows from Theorem 19 and the fact that $c_{uu}ps^2\leq(1+\epsilon)(n-1)p^2s^4\leq\frac{4}{3}np^2s^4<1$; (d) holds as $c_{uu}-1\geq(1-\epsilon)(n-1)ps^2-1\geq\frac{2}{3}(n-1)ps^2-1\geq\frac{7}{12}nps^2$.

Furthermore, note that χ_j depends on A_1 and B_1 only through the set of entries $S_j\triangleq\{\{i,j\}:i\in C(u,u)\setminus\{v\}\}$. Since $S_j\cap S_{j'}=\emptyset$ for all $j,j'\in F\setminus(J_u\cup J_v)$, it follows that χ_j's are mutually independent. Therefore,

$$\begin{aligned}
&\mathbb{P}\left\{\sum_{j\in F\setminus(J_u\cup J_v)}A_2(u,j)B_2\left(u,j\right)\leq l_{\min}\mid Q_{uv}\right\}\\
\leq&\mathbb{P}\left\{\sum_{j\in F\setminus(J_u\cup J_v)}\chi_j\leq l_{\min}\mid Q_{uv}\right\}\\
\leq&\mathbb{P}\left\{\mathrm{Binom}\left(n_R,\frac{7}{24}np^2s^4\right)\leq l_{\min}\mid Q_{uv}\right\}\leq n^{-\frac{15}{4}},
\end{aligned}\tag{44}$$

where $l_{\min}=\frac{7}{24}(\beta-6ps)n^2p^2s^4-\sqrt{\frac{35}{16}n^2\beta p^2s^4\log n}-\frac{5}{2}\log n$, and the last inequality follows from Corollary 16 with $\gamma=\frac{15}{4}\log n$ and $n(\beta-6ps)\leq n_R\leq n\beta$.

Next, we count the contribution to $W_2(u,u)$ by the incorrect seeds. Fix an incorrect seed $(j,\pi(j))$ where $j\in[n]\setminus(F\cup J_u\cup J_v)$. Note that $A_2(u,j)$ depends on A_1 through the set of entries given by $T_j\triangleq\{\{i,j\}:i\in N^{G_1}(u)\}$ and $B_2(u,\pi(j))$ depends on B_1 through the set of entries given by $\widetilde{T}_{\pi(j)}\triangleq\{\{i,\pi(j)\}:i\in N^{G_2}(u)\}$. Thus $A_2(u,j)$ and $B_2(u,\pi(j))$ are independent when $T_j\cap\widetilde{T}_{\pi(j)}=\emptyset$, which occurs if and only $j\notin N^{G_2}(u)$ or $\pi(j)\notin N^{G_1}(u)$. Thus to ensure the independence between $A_2(u,j)$ and $B_2(u,\pi(j))$ in order to facilitate computing the probability of $A_2(u,j)B_2(u,\pi(j))=1$, we also exclude the set of seeds given by $\widetilde{J}_u=N^{G_2}(u)\cap\pi^{-1}\left(N^{G_1}(u)\right)$. Let $n_W\triangleq|[n]\setminus(F\cup J_u\cup J_v\cup\widetilde{J}_u)|$. Since the event R_{uv} holds, it follows that $n_W\geq n(1-\beta)-9nps$. Now, for each $j\in[n]\setminus(F\cup J_u\cup J_v\cup\widetilde{J}_u)$, we have

$$\begin{aligned}
&\mathbb{P}\left\{A_2(u,j)B_2(u,\pi(j))=1\mid Q_{uv}\right\}\\
=&\mathbb{P}\left\{A_2(u,j)=1\mid Q_{uv}\right\}\times\mathbb{P}\left\{B_2(u,\pi(j))=1\mid Q_{uv}\right\}\\
=&\left(1-\mathbb{P}\left\{A_1(i,j)=0,\forall i\in N^{G_1}(u)\mid Q_{uv}\right\}\right)\left(1-\mathbb{P}\left\{B_1(i,\pi(j))=0,\forall i\in N^{G_2}(u)\mid Q_{uv}\right\}\right)\\
=&\left(1-(1-ps)^{a_{u\setminus v}}\right)\left(1-(1-ps)^{b_{u\setminus v}}\right)\triangleq\lambda,
\end{aligned}$$

where the last equality holds because $A_1(i,j)=0$ if $i=v$ as $j\notin J_v$; otherwise $A_1(i,j)\sim\mathrm{Bern}(ps)$; and similarly for $B_1(i,\pi(j))$.

Finally, note that $A_2(u,j)B_2(u,\pi(j))$ are dependent across different j and thus we cannot directly apply Bernstein's inequality. To see this, observe that conditional on Q_{uv}, $A_2(u,j)B_2(u,\pi(j))$ depends on A_1 and B_1 through the set of entries given by $T_j \cup \widetilde{T}_{\pi(j)} \triangleq U_j$. Therefore, for any pair of $j, j' \in [n] \setminus (F \cup J_u \cup J_v \cup \widetilde{J}_u)$ with $j \neq j'$, $A_2(u,j)B_2(u,\pi(j))$ and $A_2(u,j')B_2(u,\pi(j'))$ are dependent if and only if $U_j \cap U_{j'} \neq \emptyset$, which occurs if and only if $j' = \pi(j)$ or $j' = \pi^{-1}(j)$. Hence, we construct a dependency graph Γ for $\{A_2(u,j)B_2(u,\pi(j))\}$, where the maximum degree $\Delta(\Gamma)$ equals to 2. Thus, applying Theorem 18 with $K = 1$, $\sigma^2 = n_{\mathrm{W}}\lambda(1-\lambda)$, and $\gamma = 4\log n$ yields that

$$\mathbb{P}\left\{\sum_{j\in[n]\setminus F'} A_2(u,j)B_2(u,\pi(j)) \le n_{\mathrm{W}}\lambda - 5\sqrt{\frac{3}{2}n_{\mathrm{W}}\lambda(1-\lambda)\log n} - \frac{25}{2}\log n \mid Q_{uv}\right\} \le n^{-4},$$

where $F' = F \cup J_u \cup J_v \cup \widetilde{J}_u$.

Since $n(1-\beta-9ps) \le n_{\mathrm{W}} \le n$, we have

$$n_{\mathrm{W}}\lambda(1-\lambda) \le n\left(1-(1-ps)^{a_u}\right)\left(1-(1-ps)^{b_u}\right) \overset{(a)}{\le} na_ub_up^2s^2 \overset{(b)}{\le} \frac{9}{4}n^3p^4s^4,$$

where (a) holds as $(1+x)^r \ge 1+rx$ for every integer $r \ge 0$ and every real number $x \ge -2$; (b) holds because $a_u, b_u \le (1+\epsilon)(n-1)ps \le \frac{3}{2}nps$ under event R_{uv}. And

$$n_{\mathrm{W}}\lambda \ge n(1-\beta)\lambda - 9nps\lambda \ge n(1-\beta)\lambda - 21n^3p^5s^5.$$

Therefore, recalling that $m_{\min} = n(1-\beta)\lambda - 21n^3p^5s^5 - \frac{15}{2}\sqrt{\frac{3}{2}n^3p^4s^4\log n} - \frac{25}{2}\log n$, we get that

$$m_{\min} \le n_{\mathrm{W}}\lambda - 5\sqrt{\frac{3}{2}n_{\mathrm{W}}\lambda(1-\lambda)\log n} - \frac{25}{2}\log n.$$

It follows that

$$\mathbb{P}\left\{\sum_{j\in[n]\setminus(F\cup J_u\cup J_v\cup\widetilde{J}_u)} A_2(u,j)B_2(u,\pi(j)) \le m_{\min} \mid Q_{uv}\right\} \le n^{-4}. \tag{45}$$

Combining (43), (44), (45) with a union bound, we get that

$$\mathbb{P}\left\{W_2(u,u) \le l_{\min} + m_{\min} \mid Q_{uv}\right\} \cdot \mathbb{1}\left(R_{uv}\right) \le n^{-\frac{15}{4}} + n^{-4} < n^{-\frac{7}{2}}.$$

Remark 20 *In (44), we bound $A_2(u,j)B_2(u,j)$ from below by χ_j, by neglecting the case that j is connected to different 1-hop neighbors of u in G_1 and G_2. This lower bound is relatively tight, because*

$$\begin{aligned}\mathbb{P}\left\{A_2(u,j)B_2(u,j) = 1, \chi_j = 0 \mid Q_{uv}\right\} &\approx \mathbb{P}\left\{A_2(u,j) = 1 \mid Q_{uv}\right\}\mathbb{P}\left\{B_2(u,j) = 1 \mid Q_{uv}\right\} \\ &\approx a_ub_up^2s^2 \le \frac{9}{4}n^2p^4s^4,\end{aligned}$$

which is much smaller than $\mathbb{P}\left\{\chi_j = 1 \mid Q_{uv}\right\}$ when $np^2 \le \frac{1}{\log n}$.

E.3 Proof of Lemma 9

Fixing any two vertices $u \neq v$, we condition on Q_{uv} such that event R_{uv} holds. Note that

$$W_2(u,v) = \sum_{j=1}^{n} A_2(u,j) B_2\left(v, \pi(j)\right),$$

where A_2 and B_2 are the 2-hop adjacency matrix of G_1 and G_2, respectively. Let

$$J_0 = N^{G_1}(u) \cup \{u\} \cup \pi^{-1}\left(N^{G_2}(v)\right) \cup \pi^{-1}(v)$$

Then $A_2(u,j)B_2\left(v,\pi(j)\right) = 0$ for all $j \in J_0$. Thus,

$$W_2(u,v) = \sum_{j \in [n] \setminus J_0} A_2(u,j) B_2\left(v, \pi(j)\right).$$

Note that we have conditioned on the 1-hop neighborhoods of u and v in G_1 and G_2. In either G_1 or G_2, if u and v are connected, then a 1-hop neighbor of u (or v) may automatically become the 2-hop neighbor of v (or u). Hence, if j is connected to v in G_1 or $\pi(j)$ is connected to u in G_2, then conditioning on Q_{uv} can change the probability that $A_2(u,j)B_2\left(v,\pi(j)\right) = 1$. To circumvent this issue, we further divide the remaining seeds into five types depending on whether $j \in N^{G_1}(v) \cup \{v\}$ and $\pi(j) \in N^{G_2}(u) \cup \{u\}$, and get

$$W_2(u,v) = \sum_{k=1}^{5} \sum_{j \in J_k} A_2(u,j) B_2\left(v, \pi(j)\right). \tag{46}$$

Let $X_k = \sum_{j \in J_k} A_2(u,j) B_2\left(v,\pi(j)\right)$ denote the contribution from type k. In the sequel, we will separately bound X_k from the above for each $k \in [5]$.

Type 1: $J_1 \triangleq \{v, \pi^{-1}(u)\} \setminus J_0$. We have $X_1 \leq |J_1| \leq 2$.

Type 2: $J_2 \triangleq N^{G_1}(v) \cap \pi^{-1}\left(N^{G_2}(u)\right) \setminus J_0$. For $j \in J_2$, since $A_1(v,j) = 1$ and $B_1(u,\pi(j)) = 1$, it follows that $(j,\pi(j))$ is a 1-hop witness for (v,u). Thus, we have $X_2 \leq |J_2| \leq W_1(v,u) \leq \psi_{\max}$ on event R_{uv}.

Type 3: $J_3 \triangleq N^{G_1}(v) \setminus \left(\pi^{-1}\left(N^{G_2}(u)\right) \cup \{\pi^{-1}(u)\} \cup J_0\right)$. We have $|J_3| \leq a_v \leq \frac{3}{2}nps$ on event R_{uv}. By definition, $A_2(u,j)B_2(v,\pi(j)) \leq B_2(v,\pi(j))$. Moreover,

$$\begin{aligned}
\mathbb{P}\left\{B_2(v,\pi(j)) = 1 \mid Q_{uv}\right\} =& \mathbb{P}\left\{B_1(i,\pi(j)) = 1, \exists i \in N^{G_2}(v) \mid Q_{uv}\right\} \\
=& 1 - \mathbb{P}\left\{B_1(i,\pi(j)) = 0, \forall i \in N^{G_2}(v) \mid Q_{uv}\right\} \\
\overset{(a)}{=}& 1 - (1-ps)^{b_{v \setminus u}} \overset{(b)}{\leq} b_v ps \overset{(c)}{\leq} \frac{3}{2} np^2 s^2,
\end{aligned} \tag{47}$$

where (a) holds because $B_1(i,\pi(j)) = 0$ if $i = u$ as $\pi(j) \notin N^{G_2}(u)$; otherwise $B_1(i,\pi(j)) \overset{\text{i.i.d.}}{\sim} \text{Bern}(ps)$ across different i; (b) follows from $(1+x)^r \geq 1 + rx$ for every integer $r \geq 0$ and every real number $x \geq -2$; (c) holds due to $b_v < \frac{3}{2}nps$ on event R_{uv}.

Note that $B_2(v,\pi(j))$ only depends on B_1 through the set of entries $U_{\pi(j)} \triangleq \{\{i,\pi(j)\} : i \in N^{G_2}(v)\}$. Since $U_{\pi(j)} \cap U_{\pi(j')} = \emptyset$ for all $j \neq j' \notin J_0$, it follows that $B_2(v,\pi(j))$ are mutually independent across $j \in J_3$. Therefore,

$$\begin{aligned}
&\mathbb{P}\left\{X_3 \geq \frac{9}{2}n^2p^3s^3 + 5\log n \mid Q_{uv}\right\} \\
&\leq \mathbb{P}\left\{\sum_{j\in J_3} B_2(v,\pi(j)) \geq \frac{9}{2}n^2p^3s^3 + 5\log n \mid Q_{uv}\right\} \\
&\leq \mathbb{P}\left\{\mathrm{Binom}\left(|J_3|, \frac{3}{2}np^2s^2\right) \geq \frac{9}{2}n^2p^3s^3 + 5\log n \mid Q_{uv}\right\} \\
&\leq n^{-\frac{15}{4}},
\end{aligned} \tag{48}$$

where the last inequality follows from Corollary 16 with $\gamma = \frac{15}{4}\log n$ and $|J_3| \leq \frac{3}{2}nps$.

Type 4: $J_4 \triangleq \pi^{-1}\left(N^{G_2}(u)\right) \setminus \left(N^{G_1}(v) \cup \{v\} \cup J_0\right)$. Following the similar proof as in Type 3, we can get

$$\mathbb{P}\left\{X_4 \geq \frac{9}{2}n^2p^3s^3 + 5\log n \mid Q_{uv}\right\} \leq n^{-\frac{15}{4}}. \tag{49}$$

Type 5: $j \in J_5 \triangleq [n] \setminus \left(\cup_{k=0}^4 J_k\right)$. This is the major type. We bound X_5 by separately considering the correct and incorrect seeds.

Correct Seeds in Type 5: Recall that $F = \{j : \pi(j) = j\}$ corresponds to the set of correct seeds. We have $|F \cap J_5| \leq |F| = n\beta$. Note that $A_2(u,j)$ depends on A_1 through the set of entries given by $T_j \triangleq \{\{i,j\} : i \in N^{G_1}(u)\}$ and $B_2(v,j)$ depends on B_1 through the set of entries given by $\widetilde{T}_j \triangleq \{\{i,j\} : i \in N^{G_2}(v)\}$. Thus $A_2(u,j)$ and $B_2(v,j)$ are dependent on each other because $T_j \cap \widetilde{T}_j = \{\{i,j\} : i \in C(u,v)\} \neq \emptyset$. Thus, we bound $\mathbb{P}\{A_2(u,j)B_2(v,j) = 1\}$ by separately considering whether j is connected to some vertices in $C(u,v)$. Specifically, on the one hand,

$$\begin{aligned}
&\mathbb{P}\{\{A_2(u,j)B_2(v,j) = 1\} \cap \{A_1(i,j)B_1(i,j) = 1, \exists i \in C(u,v)\} \mid Q_{uv}\} \\
\leq&\mathbb{P}\{A_1(i,j)B_1(i,j) = 1, \exists i \in C(u,v)\} \mid Q_{uv}\} \\
=&1 - \mathbb{P}\{A_1(i,j)B_1(i,j) = 0, \forall i \in C(u,v)\} \mid Q_{uv}\} \\
\overset{(a)}{=}&1 - (1-ps^2)^{c_{uv}} \\
\overset{(b)}{\leq}&1 - (1 - c_{uv}ps^2) \overset{(c)}{\leq} \psi_{\max}ps^2,
\end{aligned} \tag{50}$$

where (a) holds because $A_1(i,j)B_1(i,j) \overset{\text{i.i.d.}}{\sim} \mathrm{Bern}(ps^2)$; (b) follows from $(1+x)^r \geq 1+rx$ for every integer $r \geq 0$ and every real number $x \geq -2$; (c) holds due to $c_{uv} < \psi_{\max}$ on event R_{uv}.

On the other hand, letting $\mathcal{A}_u(X) \triangleq \{\exists i \in X : A_1(i,j) = 1\}$,

$$\mathbb{P}\{\{A_2(u,j)B_2(v,j) = 1\} \cap \{A_1(i,j)B_1(i,j) = 0, \forall i \in C(u,v)\} \mid Q_{uv}\}$$

$$
\begin{aligned}
=&\mathbb{P}\left\{\bigcup_{X\in N^{G_1}(u)}\mathcal{A}_u(X)\cap\{\exists i\in N^{G_2}(u)\setminus X: B_1(i,j)=1\}\mid Q_{uv}\right\}\\
\le&\sum_{X\in N^{G_1}(u)}\mathbb{P}\left\{\mathcal{A}_u(X)\cap\{\exists i\in N^{G_2}(u)\setminus X: B_1(i,j)=1\}\mid Q_{uv}\right\}\\
\overset{(a)}{=}&\sum_{X\in N^{G_1}(u)}\mathbb{P}\left\{\mathcal{A}_u(X)\mid Q_{uv}\right\}\mathbb{P}\left\{\exists i\in N^{G_2}(u)\setminus X: B_1(i,j)=1\mid Q_{uv}\right\}\\
\le&\mathbb{P}\left\{\exists i\in N^{G_2}(u): B_1(i,j)=1\mid Q_{uv}\right\}\sum_{X\in N^{G_1}(u)}\mathbb{P}\left\{\mathcal{A}_u(X)\mid Q_{uv}\right\}\\
\le&\mathbb{P}\left\{A_1(i,j)=1,\exists i\in N^{G_1}(u)\mid Q_{uv}\right\}\times\mathbb{P}\left\{B_1(i,j)=1,\exists i\in N^{G_2}(v)\mid Q_{uv}\right\}\overset{(b)}{\le}\frac{9}{4}n^2p^4s^4,
\end{aligned}
\tag{51}
$$

where the equality (a) holds as $\mathcal{A}_u(X)$ and $\{\exists i\in N^{G_2}(u)\setminus X: B_1(i,j)=1\}$ are independent conditional on X. This is because $\mathcal{A}_u(X)$ depends on $T_j'\triangleq\{\{i,j\}: i\in X\}$, which is disjoint from $\widetilde{T}_j=\{\{i,j\}: i\in N^{G_2}(v)\setminus X\}$; (b) follows from the similar reasoning as in (47).

Thus, by taking the union bound over (50) and (51), we have

$$
\mathbb{P}\left\{A_2(u,j)B_2(v,j)=1\mid Q_{uv}\right\}\le\psi_{\max}ps+\frac{9}{4}n^2p^4s^4\triangleq\mu_1.
$$

Note that $A_2(u,j)B_2(v,j)$ only depends on A_1 and B_1 only through the set of entries $T_j\cup\widetilde{T}_j\triangleq U_j$. Since $U_j\cap U_{j'}=\emptyset$ for all $j\neq j'\notin J_0$, it follows that $A_2(u,j)B_2(v,j)$ are mutually independent for all $j\in F\cap J_5$. Therefore,

$$
\begin{aligned}
&\mathbb{P}\left\{\sum_{j\in F\cap J_5}A_2(u,j)B_2(v,j)\ge x_{\max}+5\log n\mid Q_{uv}\right\}\\
\le&\mathbb{P}\left\{\mathrm{Binom}(n\beta,\mu_1)\ge x_{\max}+5\log n\mid Q_{uv}\right\}\le n^{-\frac{15}{4}},
\end{aligned}
\tag{52}
$$

where $x_{\max}=2n\beta\left(\psi_{\max}ps+\frac{9}{4}n^2p^4s^4\right)$ and the last inequality follows from Corollary 16 with $\gamma=\frac{15}{4}\log n$.

Incorrect Seeds in Type 5: Let $\overline{F}\triangleq[n]\setminus F$ denote the complement of F in $[n]$. Then, $\overline{F}$ corresponds to the set of incorrect seeds with $\left|\overline{F}\right|=n(1-\beta)$. Note that $A_2(u,j)$ depends on A_1 through the set of entries given by $T_j\triangleq\{\{i,j\}: i\in N^{G_1}(u)\}$ and $B_2(v,\pi(j))$ depends on B_1 through the set of entries given by $\widetilde{T}_{\pi(j)}\triangleq\{\{i,\pi(j)\}: i\in N^{G_2}(v)\}$. Thus $A_2(u,j)$ and $B_2(v,\pi(j))$ are independent when $T_j\cap\widetilde{T}_{\pi(j)}=\emptyset$, which occurs if and only if $j\notin N^{G_2}(v)$ or $\pi(j)\notin N^{G_1}(u)$. We define $\widetilde{J}=N^{G_2}(v)\cap\pi^{-1}\left(N^{G_1}(u)\right)$, and have $\left|\widetilde{J}\right|\le b_v\le\frac{3}{2}nps$ under the event R_{uv}. Then, we separately consider the incorrect seeds depending on whether $j\in\widetilde{J}$.

- For $j\in\overline{F}\cap J_5\setminus\widetilde{J}$,

$$
\mu_2\triangleq\mathbb{P}\left\{A_2(u,j)B_2(v,\pi(j))=1\mid Q_{uv}\right\}
$$

$$
\begin{aligned}
&= \mathbb{P}\left\{A_2(u,j)=1 \mid Q_{uv}\right\} \times \mathbb{P}\left\{B_2(v,\pi(j))=1 \mid Q_{uv}\right\} \\
&= \left(1-(1-ps)^{a_{u\setminus v}}\right)\left(1-(1-ps)^{b_{v\setminus u}}\right) \le \frac{9}{4}n^2p^4s^4,
\end{aligned}
$$

where the last two steps follow from the similar reasoning as in (47).

- For $j \in \overline{F} \cap J_5 \cap \widetilde{J}$, we divide the analysis into two cases depending on whether $A_1(j,\pi(j))=1$. On the one hand,

$$
\begin{aligned}
&\mathbb{P}\left\{\{A_2(u,j)B_2(v,\pi(j)=1\} \cap \{A_1(j,\pi(j))=1\} \mid Q_{uv}\right\} \\
\le &\mathbb{P}\left\{\{A_1(j,\pi(j))=1\} \mid Q_{uv}\right\} \le ps.
\end{aligned}
$$

  On the other hand, letting $\mathcal{A}'_u \triangleq \{\exists i \in N^{G_1}(u) \setminus \{\pi(j)\} : A_1(i,j)=1\}$,

$$
\begin{aligned}
&\mathbb{P}\left\{\{A_2(u,j)B_2(v,\pi(j)=1\} \cap \{A_1(j,\pi(j))=0\} \mid Q_{uv}\right\} \\
&= \mathbb{P}\left\{\mathcal{A}'_u \cap \{B_2(v,j)=1\} \mid Q_{uv}\right\} \\
&\overset{(a)}{=} \mathbb{P}\left\{\mathcal{A}'_u \mid Q_{uv}\right\} \times \mathbb{P}\left\{\{B_2(v,j)=1\} \mid Q_{uv}\right\} \\
&\le \mathbb{P}\left\{A_1(i,j)=1, \exists i \in N^{G_1}(u) \mid Q_{uv}\right\} \times \mathbb{P}\left\{B_1(i,j)=1, \exists i \in N^{G_2}(v) \mid Q_{uv}\right\} \\
&\overset{(b)}{\le} \frac{9}{4}n^2p^4s^4,
\end{aligned}
$$

  where the equality (a) holds as A_u and $\{B_2(v,\pi(j))=1\}$ are independent. This is because $\mathcal{A}_u$ depends on $T'_j \triangleq \{\{i,j\} : i \in N^{G_1}(u) \setminus \{\pi(j)\}\}$, which is disjoint with $\widetilde{T}_{\pi(j)} = \{\{i,\pi(j)\} : i \in N^{G_2}(v)\}$; (b) follow from the similar proof in (47).

  Combining the last two displayed equations yields that

$$
\mu_3 \triangleq \mathbb{P}\left\{A_2(u,j)B_2(v,\pi(j)=1 \mid Q_{uv}\right\} \le ps + \frac{9}{4}n^2p^4s^4 \le \frac{2}{3}np^2s^2.
$$

Note that $A_2(u,j)B_2(v,\pi(j))$ are dependent across different $j \in \overline{F} \cap J_5$ and thus we cannot directly apply Bernstein's inequality. To see this, observe that conditional on Q_{uv}, $A_2(u,j)B_2(v,\pi(j))$ depends on A_1 and B_1 through the set of entries given by $T_j \cup \widetilde{T}_{\pi(j)} \triangleq U_j$. Therefore, for any pair of $j, j' \in \overline{F} \cap J_5$ with $j \neq j'$, $A_2(u,j)B_2(v,\pi(j))$ and $A_2(u,j')B_2(v,\pi(j'))$ are dependent if and only if $U_j \cap U_{j'} \neq \emptyset$, which occurs if and only if $j' = \pi(j)$ or $j' = \pi^{-1}(j)$. Hence, we construct a dependency graph Γ for $\{A_2(u,j)B_2(v,\pi(j))\}$, where the maximum degree $\Delta(\Gamma)$ equals to 2. Thus, applying Theorem 18 with $K=1$, $\sigma^2 = \left|\overline{F} \cap J_5 \setminus \widetilde{J}\right| \mu_2(1-\mu_2) + \left|\overline{F} \cap J_5 \cap \widetilde{J}\right| \mu_3(1-\mu_3)$, and $\gamma = 4\log n$ yields that

$$
\mathbb{P}\left\{\sum_{j \in \overline{F} \cap J_5} A_2(u,j)B_2(v,\pi(j)) \ge y_{\max} + \frac{25}{2}\log n \mid Q_{uv}\right\} \le n^{-4}, \tag{53}
$$

where $y_{\max} = n(1-\beta)\mu_2 + n^2p^3s^3 + \frac{5}{2}\sqrt{15n^3p^4s^4\log n}$.

Finally, combining all types of seeds and taking an union bound on (48), (49), (52) and (53), we get

$$\begin{aligned}&\mathbb{P}\left\{W_2(u,v) \geq x_{\max} + y_{\max} + 2z_{\max} + \psi_{\max} + 28\log n \mid Q_{uv}\right\} \cdot \mathbb{1}\left(R_{uv}\right)\\ \leq & 3 \cdot n^{-\frac{15}{4}} + n^{-4} < n^{-\frac{7}{2}},\end{aligned}$$

where $z_{\max} = \frac{9}{2}n^2p^3s^3$.

E.4 Proof of Lemma 11

Recall that $d_u = \left|N^{G_0}(u)\right| \sim \text{Binom}(n-1,p)$. In view of assumption $nps^2 \geq 128\log n$, applying Lemma 14 gives that

$$\mathbb{P}\left\{d_u \geq \frac{4}{3}(n-1)p\right\} \leq n^{-4}. \tag{54}$$

Let R_u denote the event $\left\{d_u < \frac{4}{3}(n-1)p\right\}$.

For any two vertices $u, v \in [n]$ with $u \neq v$, let E_{uv} denote

$$E_{uv} = \left\{N^{G_0}(u), N^{G_0}(v)\right\}.$$

Conditioning on E_{uv} such that R_u and R_v are true, we separately consider two cases: $d_u \leq d_v$ and $d_u > d_v$.

Case 1: $d_u \leq d_v$. By definition, we have $a_u \sim \text{Binom}\left(d_u, s\right)$ and $a_v \sim \text{Binom}\left(d_v, s\right)$. Applying with Bernstein's inequality given in Theorem 15 with $\gamma = \frac{15}{4}\log n$ and $K = 1$ implies that

$$\mathbb{P}\left\{a_u \geq d_u s + \sqrt{\frac{15}{2}d_u s(1-s)\log n} + \frac{5}{2}\log n \mid E_{uv}\right\} \leq n^{-\frac{15}{4}}, \tag{55}$$

$$\mathbb{P}\left\{a_v \leq d_v s - \sqrt{\frac{15}{2}d_v s(1-s)\log n} - \frac{5}{2}\log n \mid E_{uv}\right\} \leq n^{-\frac{15}{4}}. \tag{56}$$

Under the events R_u and R_v, we have

$$\begin{aligned}&d_u s + \sqrt{\frac{15}{2}d_u s(1-s)\log n} + \frac{5}{2}\log n - \left(d_v s - \sqrt{\frac{15}{2}d_v s(1-s)\log n} - \frac{5}{2}\log n\right)\\ \leq & \sqrt{\frac{15}{2}d_u s(1-s)\log n} + \sqrt{\frac{15}{2}d_v s(1-s)\log n} + 5\log n\\ \leq & 2\sqrt{10nps(1-s)} + 5\log n = \tau.\end{aligned}$$

Taking the union bound over (55) and (56), and noting that $\overline{T_{uv}} \subset \{a_u - a_v \geq \tau\}$, we have

$$\mathbb{P}\left\{\overline{T_{uv}} \mid E_{uv}\right\} \leq \mathbb{P}\left\{a_u - a_v \geq \tau \mid E_{uv}\right\} \leq n^{-\frac{15}{4}}.$$

Case 2: $d_u > d_v$. Following the similar proof, we can get

$$\mathbb{P}\left\{\overline{T_{uv}} \mid E_{uv}\right\} \le \mathbb{P}\left\{b_v - b_u \ge \tau \mid E_{uv}\right\} \le n^{-\frac{15}{4}}.$$

Combining the two cases gives that

$$\mathbb{P}\left\{\overline{T_{uv}} \mid E_{uv}\right\} \cdot \mathbb{1}\left(R_u \cap R_v\right) \le n^{-\frac{15}{4}}. \tag{57}$$

Finally, since n is sufficiently large, applying (54), (57) and the union bound yields

$$\begin{aligned}
\mathbb{P}\left\{\overline{T_{uv}}\right\} =& \mathbb{E}_{E_{uv}}\left[\mathbb{P}\left\{\overline{T_{uv}} \mid E_{uv}\right\}\right] \\
=& \mathbb{E}_{E_{uv}}\left[\mathbb{P}\left\{\overline{T_{uv}} \mid E_{uv}\right\} \cdot \mathbb{1}\left(R_u \cap R_v\right) + \mathbb{P}\left\{\overline{T_{uv}} \mid E_{uv}\right\} \cdot \mathbb{1}\left(\overline{R_u \cap R_v}\right)\right] \\
\le& \mathbb{E}_{E_{uv}}\left[\mathbb{P}\left\{\overline{T_{uv}} \mid E_{uv}\right\} \cdot \mathbb{1}\left(R_u \cap R_v\right)\right] + \mathbb{E}_{E_{uv}}\left[\mathbb{1}\left(\overline{R_u \cap R_v}\right)\right] \\
\le& n^{-\frac{15}{4}} + 2 \cdot n^{-4} \le n^{-\frac{7}{2}}.
\end{aligned}$$

E.5 Proof of Lemma 12

Since $\psi_{\max} = np^2s^2 + \sqrt{7np^2s^2 \log n} + \frac{7}{3}\log n + 2 \le 3\log n$ due to $np^2 \le \frac{1}{\log n}$, we have

$$\begin{aligned}
& l_{\min} + m_{\min} - x_{\max} - y_{\max} - 2z_{\max} - \psi_{\max} - 28\log n \\
& \ge \frac{7}{24}n^2\beta p^2 s^4 - \frac{7}{4}n^2p^3s^5 - 21n^3p^5s^5 - \sqrt{\frac{35}{16}n^2\beta p^2 s^4 \log n} - \frac{5}{2}\sqrt{15n^3p^4s^4\log n} \\
& - \frac{15}{2}\sqrt{\frac{3}{2}n^3p^4s^4\log n} - 2n\beta\left(3ps^2\log n + \frac{9}{4}n^2p^4s^4\right) - 10n^2p^3s^3 - 46\log n \\
& + n(1-\beta)\left(1-(1-ps)^{a_{u\setminus v}}\right)\left((1-ps)^{b_{v\setminus u}} - (1-ps)^{b_{u\setminus v}}\right).
\end{aligned} \tag{58}$$

To bound from below the last term in (58), we have

$$\begin{aligned}
& \left(1-(1-ps)^{a_{u\setminus v}}\right)\left((1-ps)^{b_{v\setminus u}} - (1-ps)^{b_{u\setminus v}}\right) \\
& \overset{(a)}{\ge} \left(1-(1-ps)^{a_u}\right)\left((1-ps)^{\tau} - 1\right) \\
& \overset{(b)}{\ge} -a_u\tau p^2 s^2 \\
& \overset{(c)}{\ge} -3np^3s^3\sqrt{10nps(1-s)\log n} - \frac{15}{2}np^3s^3\log n,
\end{aligned}$$

where (a) follows from $b_{v\setminus u} - b_{u\setminus v} = b_v - b_u \le \tau$ and $(1-ps)^x - (1-ps)^y \ge (1-ps)^{\tau} - 1$ when $x - y \le \tau$; (b) holds due to Bernoulli's Inequality: $(1+x)^r \ge 1 + rx$ for every integer $r \ge 0$ and every real number $x \ge -2$; (c) follows from $a_u < \frac{3}{2}nps$ and the definition of τ given in (29).

Combining the last two displayed equation gives that

$$\begin{aligned}
& l_{\min} + m_{\min} - x_{\max} - y_{\max} - 2z_{\max} - \psi_{\max} - 28\log n \\
& \ge \frac{7}{24}n^2\beta p^2 s^4 - \frac{7}{4}n^2p^3s^5 - 21n^3p^5s^5 - \sqrt{\frac{35}{16}n^2\beta p^2 s^4 \log n} - 5\sqrt{15n^3p^4s^4\log n}
\end{aligned}$$

$$
\begin{aligned}
&- 2n\beta\left(3ps^2\log n+\frac{9}{4}n^2p^4s^4\right)-10n^2p^3s^3-46\log n\\
&- 3n^2p^3s^3\sqrt{10nps(1-s)\log n}-\frac{15}{2}n^2p^3s^3\log n. \qquad (59)
\end{aligned}
$$

In view of (59), we can guarantee $l_{\min}+m_{\min}-x_{\max}-y_{\max}-2z_{\max}-\psi_{\max}-28\log n\geq 0$ if the following inequalities (60)-(65) hold. We next verify (60)-(65) hold.

By assumption that $\beta\geq 600\sqrt{\frac{\log n}{ns^4}}$, $np^2\leq\frac{1}{\log n}$, and n is sufficiently large, we have

$$
\begin{aligned}
\frac{1}{40}n^2\beta p^2s^4\geq&\frac{1}{40}n^2p^2s^4\cdot 600\sqrt{\frac{\log n}{ns^4}}\\
\geq&15n^2p^2s^2\cdot p\log n\\
\geq&n^2p^3s^3(\frac{15}{2}\log n+10+\frac{7}{4}s^2+21np^2s^2),
\end{aligned} \qquad (60)
$$

By assumption $\beta\geq\frac{600\log n}{n^2p^2s^4}$, we have

$$
\frac{1}{15}n^2\beta p^2s^4\geq\frac{1}{15}n^2\sqrt{\beta}p^2s^4\cdot\sqrt{\frac{600\log n}{n^2p^2s^4}}>\sqrt{\frac{35}{16}n^2\beta p^2s^4\log n}. \qquad (61)
$$

By assumption $\beta\geq 600\sqrt{\frac{\log n}{ns^4}}$, we have

$$
\frac{1}{30}n^2\beta p^2s^4\geq\frac{1}{30}n^2p^2s^4\cdot 600\sqrt{\frac{\log n}{ns^4}}>5\sqrt{15n^3p^4s^4\log n}. \qquad (62)
$$

By assumption $\beta\geq\frac{600\log n}{n^2p^2s^4}$, we have

$$
\frac{1}{12}n^2\beta p^2s^4\geq\frac{1}{12}n^2p^2s^4\cdot\frac{600\log n}{n^2p^2s^4}>46\log n. \qquad (63)
$$

By the assumption that $nps^2\geq 128\log n$, $np^2\leq\frac{1}{\log n}$, and n is sufficiently large, we have

$$
\begin{aligned}
\frac{1}{15}n^2\beta p^2s^4\geq&\frac{1}{20}n\beta ps^2\cdot 128\log n+\frac{1}{60}n^2\beta p^2s^4\cdot np^2\log n\\
\geq&2n\beta\left(3ps^2\log n+\frac{9}{4}n^2p^4s^4\right),
\end{aligned} \qquad (64)
$$

By the assumption $\beta\geq 600\sqrt{\frac{np^3(1-s)\log n}{s}}$, we have

$$
\frac{1}{60}n^2\beta p^2s^4\geq\frac{1}{60}n^2p^2s^4\cdot 600\sqrt{\frac{np^3(1-s)\log n}{s}}\geq 3n^2p^3s^3\sqrt{10nps(1-s)\log n}. \qquad (65)
$$

Thus, we arrive at $l_{\min}+m_{\min}\geq x_{\max}+y_{\max}+2z_{\max}+\psi_{\max}+28\log n$.

E.6 Proof of Theorem 2

Given any two vertices $u, v \in [n]$ with $u \neq v$, we let W_{uv} denote

$$W_{uv} = \{W_2(u,u) > W_2(u,v)\} \cup \{W_2(v,v) > W_2(u,v)\}.$$

We will prove W_{uv} happens with high probability. We condition on Q_{uv} such that the event R_{uv} is true. Then, we consider two cases: $b_v - b_u \leq \tau$ and $a_u - a_v \leq \tau$.

Case 1: $b_v - b_u \leq \tau$.

Let $w_{\min} \triangleq l_{\min} + m_{\min}$ and $w_{\max} \triangleq x_{\max} + y_{\max} + 2z_{\max} + \psi_{\max} + 28\log n$. According to Lemma 7 and Lemma 9, $W_2(u,u) > w_{\min}$ with high probability, and $W_2(u,v) < w_{\max}$ with high probability. Since $w_{\min} \geq w_{\max}$ according Lemma 12, we get that $W_2(u,u) > W_2(u,v)$ with high probability. More precisely, if R_{uv} occurs,

$$\begin{aligned}
&\mathbb{P}\{W_2(u,u) \leq W_2(u,v) \mid Q_{uv}\} \\
\overset{(a)}{\leq} &\mathbb{P}\{W_2(u,u) \leq w_{\min} \mid Q_{uv}\} + \mathbb{P}\{W_2(u,v) \geq w_{\max} \mid Q_{uv}\} \overset{(b)}{\leq} 2 \cdot n^{-\frac{7}{2}},
\end{aligned}$$

where (a) is based on the union bound; (b) is based on Lemma 7 and Lemma 9.

Since $\{W_2(u,u) > W_2(u,v)\} \subset W_{uv}$, it follows that,

$$\mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \leq \mathbb{P}\{W_2(u,u) \leq W_2(u,v) \mid Q_{uv}\} \leq 2 \cdot n^{-\frac{7}{2}}.$$

Case 2: $a_u - a_v \leq \tau$.

We can lower bound $W_2(v,v)$ analogous to Lemma 7, and prove that the lower bound is no smaller than the upper bound of $W_2(u,v)$ in this case. Then,

$$\mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \leq \mathbb{P}\{W_2(v,v) \leq W_2(u,v) \mid Q_{uv}\} \leq 2 \cdot n^{-\frac{7}{2}}.$$

Since $T_{uv} = \{a_u - a_v \leq \tau\} \cup \{b_v - b_u \leq \tau\}$, applying the union bound yields that

$$\begin{aligned}
&\mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \cdot \mathbb{1}(R_{uv} \cap T_{uv}) \\
=&\mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \cdot \mathbb{1}(R_{uv}) \cdot \mathbb{1}(T_{uv}) \\
\leq&\mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \cdot \mathbb{1}(R_{uv}) \cdot \mathbb{1}(b_v - b_u \leq \tau) \\
&+ \mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \cdot \mathbb{1}(R_{uv}) \cdot \mathbb{1}(a_u - a_v \leq \tau) \leq 4 \cdot n^{-\frac{7}{2}}.
\end{aligned}$$

Then, applying Lemma 6 and Lemma 11 yields that

$$\begin{aligned}
\mathbb{P}\left\{\overline{W_{uv}}\right\} =&\mathbb{E}_{Q_{uv}}\left[\mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \cdot \mathbb{1}(R_{uv} \cap T_{uv}) + \mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \cdot \mathbb{1}\left(\overline{R_{uv} \cap T_{uv}}\right)\right] \\
\leq&\mathbb{E}_{Q_{uv}}\left[\mathbb{P}\left\{\overline{W_{uv}} \mid Q_{uv}\right\} \cdot \mathbb{1}(R_{uv} \cap T_{uv})\right] + \mathbb{E}_{Q_{uv}}\left[\mathbb{1}\left(\overline{R_{uv} \cap T_{uv}}\right)\right] \\
\leq&6 \cdot n^{-\frac{7}{2}}.
\end{aligned}$$

Finally, applying the union bound over all pairs (u,v) with $u \neq v$, we get that

$$\mathbb{P}\left\{\bigcap_{u,v \in [n], u \neq v} W_{uv}\right\} \geq 1 - \sum_{u \neq v} \mathbb{P}\left\{\overline{W_{uv}}\right\} \geq 1 - 6 \cdot n^{-\frac{3}{2}} \geq 1 - n^{-1}.$$

Assuming $\bigcap_{u,v\in[n],u\neq v} W_{uv}$ is true, we next show that the output of GMWM, $\widetilde{\pi}$, must be equal to π^*.

We prove this by contradiction. Suppose in contrary that $\widetilde{\pi} \neq \pi^*$. Assume the first fake pair is chosen by GMWM in the k-th iteration, which implies that GMWM selects true pairs in the first $k-1$ iterations. We let $\left(u^k, v^k\right)$ denote the fake pair chosen at the k-th iteration. Because $\bigcap_{u,v\in[n],u\neq v} W_{uv}$ is true, we have $W_2\left(u^k, u^k\right) > W_2\left(u^k, v^k\right)$ or $W_2\left(v^k, v^k\right) > W_2\left(u^k, v^k\right)$. We consider two cases. The first case is that $\left(u^k, u^k\right)$ or $\left(v^k, v^k\right)$ has been selected in the first $k-1$ iterations, in which case the fake pair $\left(u^k, v^k\right)$ would have been eliminated before the k-th iteration. The second case is that $\left(u^k, u^k\right)$ and $\left(v^k, v^k\right)$ have not been selected in the first $k-1$ iterations. Then, GMWM would select one of them instead of $\left(u^k, v^k\right)$ in the k-th iteration. Thus, both cases contradict to the assumption that GMWM picks a fake pair in the k-th iteration.

Hence, GMWM outputs n true pairs. Then, we have $\mathbb{P}\left\{\widetilde{\pi} = \pi^*\right\} \geq \mathbb{P}\left\{\bigcap_{\substack{u,v\in[n]\\u\neq v}} W_{uv}\right\} \geq 1 - n^{-1}$.

Appendix F. Proof of Theorem 3

Suppose $nps^2 - \log n = c$ for $c < +\infty$. Recall that $G_1^{\pi^*}$ is the graph obtained by relabeling every vertex i in G_1 by $\pi^*(i)$, and the intersection graph $G_1^{\pi^*} \wedge G_2$ includes the common edges in both $G_1^{\pi^*}$ and G_2. Since $G_1^{\pi^*} \wedge G_2 \sim \mathcal{G}(n, ps^2)$, Bollobás (2001, Section 3.1) shows that the distribution of the number of isolated vertices in $G_1^{\pi^*} \wedge G_2$ converges to Pois(e^{-c}). Let $\mathcal{F}_1$ denote the event that there are at least three isolated vertices in $G_1^{\pi^*} \wedge G_2$. Then $\mathbb{P}\left\{\mathcal{F}_1\right\} = \Omega(1)$.

Let $\mathcal{F}_2$ denote the event that there are at least three isolated vertices that are incorrectly seeded in $G_1^{\pi^*} \wedge G_2$. Since we construct the seed set by choosing $n\beta$ correct seeds uniformly from n vertices, it follows that $\mathbb{P}\left\{\mathcal{F}_2\right\} \geq \mathbb{P}\left\{\mathcal{F}_1\right\} \frac{\binom{n-3}{n\beta}}{\binom{n}{n\beta}} = \mathbb{P}\left\{\mathcal{F}_1\right\} \frac{(n-3)!}{n!} \cdot \frac{(n(1-\beta))!}{(n(1-\beta)-3)!} = \Omega((1-\beta)^3)$.

Since the prior distribution of the ground truth permutation π^* is uniform, the maximum likelihood estimator (MLE) $\widehat{\pi}_{ML}$ is equivalent to the maximum a posterior probability (MAP) estimator, both of which minimize the error probability $\mathbb{P}\left\{\widehat{\pi} \neq \pi^*\right\}$ among all possible estimators. Below, we will show that the probability with which MLE recovers the correct matching π^* is at most $\Omega((1-\beta)^3)$.

Recall that A_1 and B_1 are the adjacency matrix for G_1 and G_2, respectively, and π is the partially-correct seed mapping. Let $\mathcal{S}(\pi)$ denote the set of all possible permutations $\widehat{\pi}$ on $\{1, 2, ..., n\}$ such that $\widehat{\pi}(i) = \pi(i)$ for exactly $n\beta$ vertices. Under the model given in Section 2, the maximum likelihood estimator $\widehat{\pi}_{ML}$ is

$$\widehat{\pi}_{ML} = \underset{\widehat{\pi}\in\mathcal{S}(\pi)}{\arg\max}\, \mathbb{P}\left\{A_1, B_1, \pi \mid \widehat{\pi}\right\}.$$

We next prove that $\widehat{\pi}_{ML}$ may not be π^*. Our key idea is to show that there exist multiple $\widetilde{\pi} \in \mathcal{S}(\pi)$ that differ from π^*, with

$$\mathbb{P}\left\{A_1, B_1, \pi \mid \widetilde{\pi}\right\} \geq \mathbb{P}\left\{A_1, B_1, \pi \mid \pi^*\right\}.$$

Towards this end, we first derive an expression for $\mathbb{P}\{A_1, B_1, \pi \mid \widehat{\pi}\}$ given that the true mapping is $\widehat{\pi}$ in $\mathcal{S}(\pi)$. Note that

$$\mathbb{P}\{A_1, B_1, \pi \mid \widehat{\pi}\} = \mathbb{P}\{\pi \mid \widehat{\pi}\}\,\mathbb{P}\{A_1, B_1 \mid \pi, \widehat{\pi}\} = \mathbb{P}\{\pi \mid \widehat{\pi}\}\,\mathbb{P}\{A_1, B_1 \mid \widehat{\pi}\}, \tag{66}$$

where the last equality holds because the seed mapping π is generated independently from the graphs G_1 and G_2. For any $\widehat{\pi} \in \mathcal{S}(\pi)$, since $\widehat{\pi}$ and π coincide at exactly $n\beta$ vertices, we can get

$$\mathbb{P}\{\pi \mid \widehat{\pi}\} = \frac{1}{\binom{n}{n\beta}\cdot !(n(1-\beta))} = \mathbb{P}\{\pi \mid \pi^*\}, \tag{67}$$

where $!k$ is the k-th derangement number. A derangement is a permutation of the elements of a set, such that no element appears in its original position. The k-th derangement number is the total number of such derangements when the set has k elements.

For $\mathbb{P}\{A_1, B_1 \mid \widehat{\pi}\}$, by definition of the correlated Erdős-Rényi model, we have

$$\mathbb{P}\{A_1(i,j), B_1(\widehat{\pi}(i), \widehat{\pi}(j)) \mid \widehat{\pi}\} = \begin{cases} ps^2 & \text{if } A_1(i,j) = B_1(\widehat{\pi}(i), \widehat{\pi}(j)) = 1, \\ ps(1-s) & \text{if } A_1(i,j) + B_1(\widehat{\pi}(i), \widehat{\pi}(j)) = 1, \\ 1 - p + p(1-s)^2 & \text{if } A_1(i,j) = B_1(\widehat{\pi}(i), \widehat{\pi}(j)) = 0. \end{cases}$$

This formula can be rewritten as

$$\begin{aligned} \log \mathbb{P}\{A_1(i,j), B_1(\widehat{\pi}(i), \widehat{\pi}(j)) \mid \widehat{\pi}\} =& A_1(i,j)B_1(\widehat{\pi}(i), \widehat{\pi}(j)) \log \frac{ps^2(1-2ps+ps^2)}{p^2s^2(1-s)^2} \\ &+ (A_1(i,j) + B_1(\widehat{\pi}(i), \widehat{\pi}(j))) \log \frac{ps(1-s)}{1-2ps+ps^2} \\ &+ \log(1-2ps+ps^2). \end{aligned}$$

Since $\{A_1(i,j), B_1(\widehat{\pi}(i), \widehat{\pi}(j))\}$ is independent across different pairs of vertices, we have

$$\begin{aligned} \log \mathbb{P}\{A_1, B_1 \mid \widehat{\pi}\} =& \sum_{1 \le i < j \le n} \log \mathbb{P}\{A_1(i,j), B_1(\widehat{\pi}(i), \widehat{\pi}(j)) \mid \widehat{\pi}\} \\ =& \frac{1}{2}\langle A_1, \widehat{\Pi} B_1 \widehat{\Pi}^T \rangle \log \frac{ps^2(1-2ps+ps^2)}{p^2s^2(1-s)^2} \\ &+ (\mathbf{1}^T A_1 \mathbf{1} + \mathbf{1}^T B_1 \mathbf{1}) \log \frac{ps(1-s)}{1-2ps+ps^2} \\ &+ (n^2 - n)\log(1-2ps+ps^2), \end{aligned} \tag{68}$$

where $\widehat{\Pi}$ is the permutation matrix corresponding to $\widehat{\pi}$, $\langle \cdot, \cdot \rangle$ is the Frobenius inner product, and $\mathbf{1}$ is a $n \times 1$ vector whose entries are all 1's. Because A_1 and B_1 are observed and fixed, maximizing $\log \mathbb{P}\{A_1, B_1 \mid \widehat{\pi}\}$ is equivalent to maximizing $\langle A_1, \widehat{\Pi} B_1 \widehat{\Pi}^T \rangle$ over all permutations $\widehat{\pi} \in \mathcal{S}(\pi)$.

Based on (68), we can now specify the alternative $\widetilde{\pi} \in \mathcal{S}(\pi)$ such that $\mathbb{P}\{A_1, B_1 \mid \widetilde{\pi}\} \ge \mathbb{P}\{A_1, B_1 \mid \pi^*\}$. Let I denote the union of the set of correct seeds and the set of all non-isolated vertices in $G_1^{\pi^*} \wedge G_2$. Then, its compliment I^c is the set of isolated vertices that

are incorrectly seeded in $G_1^{\pi^*} \wedge G_2$. Let $\widetilde{\mathcal{S}}(\pi)$ denote the set of all possible permutations $\widetilde{\pi} \in \mathcal{S}(\pi)$ such that $\widetilde{\pi}$ coincides with π^* on the set I, i.e., $\widetilde{\pi}(i) = \pi^*(i)$ for $i \in I$. Then, we must have $\pi^* \in \widetilde{\mathcal{S}}(\pi) \subset \mathcal{S}(\pi)$. Note that for any $\widetilde{\pi} \in \widetilde{\mathcal{S}}(\pi) \subset \mathcal{S}(\pi)$, we have

$$\begin{aligned}\langle A_1, \widetilde{\Pi} B_1 \widetilde{\Pi}^T \rangle &\geq \sum_{(i,j)\in I\times I} A_1(i,j) B_1(\widetilde{\pi}(i), \widetilde{\pi}(j)) \\ &\overset{(a)}{=} \sum_{(i,j)\in I\times I} A_1(i,j) B_1(\pi^*(i), \pi^*(j)) \\ &\overset{(b)}{=} \sum_{(i,j)\in [n]\times [n]} A_1(i,j) B_1(\pi^*(i), \pi^*(j)), \end{aligned} \tag{69}$$

where (a) follows from $\widetilde{\pi}(i) = \pi^*(i)$ for $i \in I$, and (b) holds due to $A_1(i,j)B_1(\pi^*(i), \pi^*(j)) = 0$ for all $(i,j) \notin I \times I$.

Combining (66), (67) and (69), for any $\widetilde{\pi} \in \widetilde{\mathcal{S}}(\pi)$, we have

$$\begin{aligned}\mathbb{P}\{A_1, B_1, \pi \mid \widetilde{\pi}\} =& \mathbb{P}\{A_1, B_1 \mid \widetilde{\pi}\} \mathbb{P}\{\pi \mid \widetilde{\pi}\} \\ \geq& \mathbb{P}\{A_1, B_1 \mid \pi^*\} \mathbb{P}\{\pi \mid \pi^*\} \\ =& \mathbb{P}\{A_1, B_1, \pi \mid \pi^*\}. \end{aligned}$$

It remains to count the number of such $\widetilde{\pi} \in \widetilde{\mathcal{S}}(\pi)$. Towards this end, we first show below that every derangement of I^c corresponds to a distinct $\widetilde{\pi}$. Let $X = \{k : k = \pi(i), i \in I^c\}$ and $Y = \{k : k = \pi^*(i), i \in I^c\}$. Note that $|X| = |Y| = |I^c|$. We let $f : X \to Y$ denote an injective mapping such that $f(x) = x$ for every $x \in X \cap Y$. We thus have $Y = \{k : k = f(\pi(i)), i \in I^c\}$. Then, let π' be a permutation on $\{1, 2, ..., n\}$ such that $\pi'(i) = \pi^*(i)$ for $i \in I$ and $\pi'(i) \neq f(\pi(i))$ for $i \in I^c$. Thus, $\{k : k = \pi'(i), i \in I^c\}$ must be a derangement of $Y = \{k : k = f(\pi(i)), i \in I^c\}$. We can conclude that every derangement of I^c corresponds to a distinct π'. It only remains to show that every such $\pi' \in \widetilde{\mathcal{S}}$. First, for $i \in I^c$, we have $\pi'(i) \neq \pi(i)$. This is because, (i) if $\pi(i) \in X \cap Y$, then $\pi'(i) \neq \pi(i)$ due to $f(\pi(i)) = \pi(i)$ and $\pi'(i) \neq f(\pi(i))$; (ii) if $\pi(i) \notin X \cap Y$, then $\pi'(i) \neq \pi(i)$ due to $\pi(i) \notin Y$. Second, we have $\pi'(i) = \pi^*(i)$ for $i \in I$. Thus, we must have $\pi' \in \widetilde{\mathcal{S}}$. In summary, every derangement of I^c corresponds to a distinct $\widetilde{\pi} \in \widetilde{\mathcal{S}}(\pi)$. By counting the number of derangement of I^c, we can then conclude that $\left|\widetilde{\mathcal{S}}(\pi)\right| \geq (!\,|I^c|)$.

Note that π^* also belongs to $\widetilde{\mathcal{S}}(\pi)$. Hence, there are at least $(!\,|I^c| - 1)$ different incorrect permutations in $\widetilde{\mathcal{S}}(\pi)$ whose likelihood is at least as large as the ground truth π^*. Thus, the MLE is correct with probability at most $1/(!\,|I^c|)$. Note that on the event $\mathcal{F}_2$, we have $|I^c| \geq 3$. Therefore, MLE is correct with probability at most $1/2$ conditioned on $\mathcal{F}_2$. In conclusion, MLE is correct with probability at most $(1/2)\mathbb{P}\{\mathcal{F}_2\} = \Omega((1-\beta)^3)$.

References

Yonathan Aflalo, Alexander Bronstein, and Ron Kimmel. On convex relaxation of graph isomorphism. *Proceedings of the National Academy of Sciences*, 112(10):2942–2947, 2015.

Boaz Barak, Chi-Ning Chou, Zhixian Lei, Tselil Schramm, and Yueqi Sheng. (Nearly) efficient algorithms for the graph matching problem on correlated random graphs. *arXiv preprint arXiv:1805.02349*, 2018.

Florian Bernard, Christian Theobalt, and Michael Moeller. Ds*: Tighter lifting-free convex relaxations for quadratic matching problems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4310–4319, 2018.

Béla Bollobás. *Random Graphs (2nd Edition)*. Cambridge Studies in Advanced Mathematics, 2001.

Carla-Fabiana Chiasserini, Michele Garetto, and Emilio Leonardi. Social network de-anonymization under scale-free user relations. *IEEE/ACM transactions on networking*, 24(6):3756–3769, 2016.

Donatello Conte, Pasquale Foggia, Carlo Sansone, and Mario Vento. Thirty years of graph matching in pattern recognition. *International journal of pattern recognition and artificial intelligence*, 18(03):265–298, 2004.

Timothee Cour, Praveen Srinivasan, and Jianbo Shi. Balanced graph matching. In *Advances in Neural Information Processing Systems*, pages 313–320, 2007.

Daniel Cullina and Negar Kiyavash. Improved achievability and converse bounds for Erdös-Rényi graph matching. In *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*, pages 63–72. ACM, 2016.

Daniel Cullina and Negar Kiyavash. Exact alignment recovery for correlated Erdös-Rényi graphs. *arXiv preprint arXiv:1711.06783*, 2017.

Daniel Cullina, Negar Kiyavash, Prateek Mittal, and H Vincent Poor. Partial recovery of Erdös-Rényi graph alignment via k-core alignment. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–21, 2019.

Osman Emre Dai, Daniel Cullina, Negar Kiyavash, and Matthias Grossglauser. On the performance of a canonical labeling for matching correlated Erdös-Rényi graphs. *arXiv preprint arXiv:1804.09758*, 2018.

Jian Ding, Zongming Ma, Yihong Wu, and Jiaming Xu. Efficient random graph matching via degree profiles. *Probability Theory and Related Fields*, 179(1):29–115, 2021.

Devdatt P. Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009.

Nadav Dym, Haggai Maron, and Yaron Lipman. DS++: a flexible, scalable and provably tight relaxation for matching problems. *ACM Transactions on Graphics (TOG)*, 36(6): 184, 2017.

Jack Edmonds and Richard M Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, 19(2):248–264, 1972.

Zhou Fan, Cheng Mao, Yihong Wu, and Jiaming Xu. Spectral graph matching and regularized quadratic relaxations I: The Gaussian model. *arxiv preprint arXiv:1907.08880*, 2019a.

Zhou Fan, Cheng Mao, Yihong Wu, and Jiaming Xu. Spectral graph matching and regularized quadratic relaxations II: Erdős-Rényi graphs and universality. *arxiv preprint arXiv:1907.08883*, 2019b.

Zhou Fan, Cheng Mao, Yihong Wu, and Jiaming Xu. Spectral graph matching and regularized quadratic relaxations: Algorithm and theory. In *International Conference on Machine Learning*, pages 2985–2995. PMLR, 2020.

Soheil Feizi, Gerald Quon, Mariana Mendoza, Muriel Medard, Manolis Kellis, and Ali Jadbabaie. Spectral alignment of graphs. *IEEE Transactions on Network Science and Engineering*, 2019.

Marcelo Fiori and Guillermo Sapiro. On spectral properties for graph matching and graph isomorphism problems. *Information and Inference: A Journal of the IMA*, 4(1):63–76, 2015.

Marcelo Fiori, Pablo Sprechmann, Joshua Vogelstein, Pablo Musé, and Guillermo Sapiro. Robust multimodal graph matching: Sparse coding meets graph matching. In *Advances in Neural Information Processing Systems*, pages 127–135, 2013.

Donniell E. Fishkind, Sancar Adali, and Carey E. Priebe. Seeded graph matching. *arXiv preprint arXiv:1209.0367*, 2018.

Luca Ganassali and Laurent Massoulié. From tree matching to sparse graph alignment. In *Conference on Learning Theory*, pages 1633–1665. PMLR, 2020.

Marco Gori, Marco Maggini, and Lorenzo Sarti. Exact and approximate graph matching using random walks. *IEEE transactions on pattern analysis and machine intelligence*, 27: 1100–11, 08 2005.

Aria D Haghighi, Andrew Y Ng, and Christopher D Manning. Robust textual inference via graph matching. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 387–394. Association for Computational Linguistics, 2005.

Svante Janson. Large deviations for sums of partly dependent random variables. *Random Struct. Algorithms*, 24:234–248, 2004.

Svante Janson, Tomasz Łuczak, Tatyana Turova, Thomas Vallier, et al. Bootstrap percolation on the random graph $G_{n,p}$. *The Annals of Applied Probability*, 22(5):1989–2047, 2012.

Ehsan Kazemi, S Hamed Hassani, and Matthias Grossglauser. Growing a graph matching from a handful of seeds. *Proceedings of the VLDB Endowment*, 8(10):1010–1021, 2015.

Ehsan Kazemi, Hamed Hassani, Matthias Grossglauser, and Hassan Pezeshgi Modarres. Proper: global protein interaction network alignment through percolation matching. *BMC bioinformatics*, 17(1):527, 2016.

Vladimir G Kim, Yaron Lipman, and Thomas Funkhouser. Blended intrinsic maps. *ACM Transactions on Graphics (TOG)*, 30(4):1–12, 2011.

Nitish Korula and Silvio Lattanzi. An efficient reconciliation algorithm for social networks. *Proceedings of the VLDB Endowment*, 7(5):377–388, 2014.

Z Lähner, Emanuele Rodolà, MM Bronstein, Daniel Cremers, Oliver Burghard, Luca Cosmo, Andreas Dieckmann, Reinhard Klein, and Y Sahillioglu. SHREC'16: Matching of deformable shapes with topological noise. *Proc. 3DOR*, 2(10.2312), 2016.

François Le Gall. Powers of tensors and fast matrix multiplication. In *Proceedings of the 39th international symposium on symbolic and algebraic computation*, pages 296–303, 2014.

Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June 2014.

Joseph Lubars and R Srikant. Correcting the output of approximate graph matching algorithms. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pages 1745–1753. IEEE, 2018.

Vince Lyzinski, Donniell E. Fishkind, and Carey E. Priebe. Seeded graph matching for correlated Erdős-Rényi graphs. *Journal of Machine Learning Research*, 15, 2013.

Vince Lyzinski, Donniell Fishkind, Marcelo Fiori, Joshua Vogelstein, Carey Priebe, and Guillermo Sapiro. Graph matching: Relax at your own risk. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 38(1):60–73, 2016.

Haggai Maron and Yaron Lipman. (Probably) concave graph matching. In *Advances in Neural Information Processing Systems*, pages 408–418, 2018.

Elchanan Mossel and Jiaming Xu. Seeded graph matching via large neighborhood statistics. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1005–1014. SIAM, 2019.

Arvind Narayanan and Vitaly Shmatikov. De-anonymizing social networks. In *Security and Privacy, 2009 30th IEEE Symposium on*, pages 173–187. IEEE, 2009.

Pedram Pedarsani and Matthias Grossglauser. On the privacy of anonymized networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1235–1243. ACM, 2011.

Yusuf Sahillioğlu. Recent advances in shape correspondence. *The Visual Computer*, 36(8): 1705–1721, 2020.

Christian Schellewald and Christoph Schnörr. Probabilistic subgraph matching based on convex relaxation. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 171–186. Springer, 2005.

Farhad Shirani, Siddharth Garg, and Elza Erkip. Seeded graph matching: Efficient algorithms and theoretical guarantees. In *2017 51st Asilomar Conference on Signals, Systems, and Computers*, pages 253–257. IEEE, 2017.

Rohit Singh, Jinbo Xu, and Bonnie Berger. 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.

Amanda L Traud, Peter J Mucha, and Mason A Porter. Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165–4180, 2012.

Shinji Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE transactions on pattern analysis and machine intelligence*, 10(5):695–703, 1988.

Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. In *Computer Graphics Forum*, volume 30, pages 1681–1707. Wiley Online Library, 2011.

Joshua T Vogelstein, John M Conroy, Vince Lyzinski, Louis J Podrazik, Steven G Kratzer, Eric T Harley, Donniell E Fishkind, R Jacob Vogelstein, and Carey E Priebe. Fast approximate quadratic programming for graph matching. *PLOS one*, 10(4):e0121002, 2015.

Yihong Wu, Jiaming Xu, and Sophie H Yu. Settling the sharp reconstruction thresholds of random graph matching. *arXiv preprint arXiv:2102.00082*, 2021.

Hongteng Xu, Dixin Luo, Hongyuan Zha, and Lawrence Carin. Gromov-wasserstein learning for graph matching and node embedding. *arXiv preprint arXiv:1901.06003*, 2019.

Lyudmila Yartseva and Matthias Grossglauser. On the performance of percolation graph matching. In *Proceedings of the first ACM conference on Online social networks*, pages 119–130. ACM, 2013.

Tianshu Yu, Junchi Yan, Yilin Wang, Wei Liu, and Baoxin Li. Generalizing graph matching beyond quadratic assignment model. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 861–871, 2018.

Mikhail Zaslavskiy, Francis Bach, and Jean-Philippe Vert. A path following algorithm for the graph matching problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2227–2242, 2008.

Zhen Zhang, Yijian Xiang, Lingfei Wu, Bing Xue, and Arye Nehorai. Kergm: Kernelized graph matching. In *Advances in Neural Information Processing Systems*, pages 3330–3341, 2019.