

Math 888 Final Project: Exact Matching of Random Graphs with Constant Correlation

Daniel Szabo

December 15, 2021

1 Introduction

For my final project I will be summarizing the result of Mao, Rudelson and Tikhomorov on the Exact Matching of Random Graphs with Constant Correlation in [MRT]. Starting from two correlated random graphs G, G' following the $G(n, p)$ Erdős-Rényi model with noise δ and a latent permutation π , the graph matching problem is to recover π given only G' and $G^\pi = (\pi(V), E)$. There are many regimes to study this problem depending on p, δ and the level to which we want to recover π . They prove that for a constant correlation $\delta, \varepsilon > 0$ and some constant C if $(1 + \varepsilon) \log n \leq np \leq n^{\frac{1}{C \log \log n}}$ then there is a polynomial time algorithm that recovers π exactly with probability $1 - o(1)$.

2 Background

We can now explain the above problem in more detail, as well as give a general overview of related work. First it is important to note this problem is very closely related to graph isomorphism [Bab16], one of the few remaining NP-intermediate problems, so a nice exact result in a very general case would be surprising. In general with the random noise from the Erdős-Rényi model this is an instance of quadratic assignment, which is known to be NP-hard even for the approximation version [PRW94, BCPP98].

2.1 The Correlated Erdős-Rényi model

The general Erdős-Rényi model $G(n, p)$ is a random graph on n vertices for which each edge is present independently with probability p . For the correlated version, two graphs G and G' with adjacencies A and B are sampled from $G(n, p)$ with correlated adjacency matrices following the following relations depending on the noise level δ :

$$\mathbb{P}[B_{i,j} = 0 | A_{i,j} = 1] = \delta \iff \mathbb{E}[A_{i,j} B_{i,j}] = p(1 - \delta)$$

In other words the correlation between any particular edge of G and its permuted version G^π is $1 - \delta$. As is usual for these models, to have any chance of recovering the permutation of π we need the graphs to be connected, which is almost surely the case for a single graph if $np \geq (1 + \varepsilon) \log n$,

and for the correlated version if $np(1 - \delta) \geq (1 + \varepsilon) \log n$ for n large enough. We will look at the regime where δ is a constant, so the bound $np \geq (1 + \varepsilon) \log n$ is sufficient.

2.2 Related Results

For exact recovery, this is the first algorithm that handles constant δ in polynomial time. Other results such as [DMWX21, FMWX19, MRT21] handle the case where δ (slowly) vanishes in n , and [WXY21] gives an exponential algorithm for general δ . These results are summarized in Table 1.

Table 1: Conditions for Exact Matching

	Condition	Time Complexity
[WXY21]	$np(1 - \delta) \geq (1 + \varepsilon) \log n$ if $\frac{p}{1-\delta} = o(1)$	Exponential
[BCL ⁺ 19]	$np \geq n^{o(1)}, 1 - \delta \geq (\log n)^{-o(1)}$	Quasipolynomial
[DMWX21]	$np \geq (\log n)^C, \delta \leq (\log n)^{-C}$	Polynomial
	$C \log n \leq np \leq e^{(\log \log n)^C}, \delta \leq (\log \log n)^{-C}$	
[FMWX19]	$np \geq (\log n)^C, \delta \leq (\log n)^{-C}$	Polynomial
[MRT21]	$np \geq (\log n)^C, \delta \leq (\log \log n)^{-C}$	Polynomial
[MRT]	$(1 + \varepsilon) \log n \leq np \leq n^{\frac{1}{C \log \log n}}, \delta \leq \min(\text{const}, \varepsilon/4)$	Polynomial

2.2.1 Vertex Signatures

Many of these results were proven using some variety of vertex signatures. A signature f_i^A for a vertex i in A is some value that “nearly uniquely” encodes some local information for i . The idea to recover the latent permutation π^* between A and B is then to match the signatures f_i^A to the closest f_j^B .

A naïve example would be to simply take $f_i^A = \deg_i^A$, the degree of vertex i in A . This however is not “unique” enough, as the vertex degrees are distributed with mean np and variance \sqrt{np} , so we would have too many repeated signatures as, $n \gg \sqrt{np}$, and could not recover π^* .

In the above [DMWX21] they go one step further and use degree profiles as signatures. These profiles are vectors of the degrees of all neighboring vertices, and are sufficient to solve the case for vanishing δ . For the slightly different seeded version of the problem, [MX20] use the number of r -neighbors, or the number of nodes within r steps of i , as a signature. This is sufficient even for constant δ in this case. Meanwhile for partial matching and constant δ , [GML21] use local trees of depth $O(\log n)$ as signatures. This result combines these methods by first using the degrees of some “nearby” vertices as a signature to get a partial matching, and then refining it using local degree information.

3 Results

The main theorems of the paper which I will try to sketch a proof of are the following results:

Theorem A (Almost exact matching). *For any constant $D > 0$ there are some constants $\delta_0, c, C > 0$ with the following property. Let G^π and G' be the graphs given by the correlated Erdős–Rényi graph model with parameters n, p , and δ such that for n large enough*

$$\log n \leq np(1 - \delta) \leq n^{\frac{1}{C \log \log n}}.$$

Then there is a random function F_{al} defined on pairs of graphs on $[n]$ and taking values in the set of permutations on $[n]$, such that for a

- F_{al} is independent from the graphs G^π and G' ,
- F_{al} has expected time complexity $O(n^{2+o(1)})$, and
- for any latent permutation $\pi : [n] \rightarrow [n]$,

$$\mathbb{P}\{F_{\text{al}}(G^\pi, G')(i) \neq \pi(i) \text{ for at most } n^{1-c} \text{ indices } i \in [n]\} \geq 1 - n^{-D}.$$

We can then take this almost exact matching and produce an exact matching with high probability.

Theorem B (Exact matching). *For any constant $\varepsilon \in (0, 1]$, there exist absolute constants $\delta_0, C > 0$ with the following property. Let G^π and G' be the graphs given by the correlated Erdős–Rényi graph model with parameters n, p , and δ such that*

$$(1 + \varepsilon) \log n \leq np \leq n^{\frac{1}{C \log \log n}}, \quad 0 < \delta \leq \min(\delta_0, \varepsilon/4).$$

Then there is a random function F_{ex} defined on pairs of graphs on $[n]$ and taking values in the set of permutations on $[n]$, such that

- F_{ex} is independent from the graphs G^π and G' ,
- F_{ex} has expected time complexity $n^{2+o(1)}$, and
- for every permutation $\pi : [n] \rightarrow [n]$,

$$\mathbb{P}\{F_{\text{ex}}(G^\pi, G') = \pi\} \geq 1 - n^{-10} - \exp(-\varepsilon pn/10).$$

4 Algorithms

The core part of this result is Theorem A, which depends on a good choice of signatures. This signature is dependent on the following notion of partition trees.

4.1 Partition Trees

For a graph A and vertex i , let $S(i, r)$ be the r -sphere of i with the standard graph distance, namely all the vertices reachable from i within r steps. A partition tree T is then a complete binary tree of depth $m = C \log \log n$ with nodes T_σ^r representing *sets of vertices in A* indexed by strings $\sigma \in \{\pm 1\}^r$ for $r = 1 \dots m$ constructed by Algorithm 1.

Algorithm 1 VertexSignature

Input: a graph A on the vertex set $[n]$, a vertex $i \in [n]$, and a depth parameter m

Output: a signature vector $f \in \mathbb{R}^{2^m}$

```

1:  $T_\emptyset^0 \leftarrow \{i\}$   $\triangleright \emptyset$  denotes the empty tuple
2: for  $r = 0, \dots, m-1$  do
3:   for  $\sigma \in \{-1, 1\}^k$  do
4:      $T_{(\sigma, +1)}^{r+1} \leftarrow \{j \in N(T_\sigma^k) \cap S(i, r+1) : \deg(j) \geq np\}$ 
5:      $T_{(\sigma, -1)}^{r+1} \leftarrow \{j \in N(T_\sigma^k) \cap S(i, r+1) : \deg(j) < np\}$ 
6:   end for
7: end for
8: define  $f(i) \in \mathbb{R}^{2^m}$  by  $f(i)_\sigma := \sum_{j \in N(T_\sigma^m) \cap S(i, m+1)} (\deg(j) - 1 - np)$  for  $\sigma \in \{-1, 1\}^m$ 
9: return  $f(i)$ 

```

Here $N(T_\sigma^r)$ denotes the set of all neighbors of all vertices in T_σ^r in A . In essence this algorithm constructs the partition T by splitting at each layer r between nodes with high degree and low degree to partition each layer into 2^r sets. The final signature comes from some degree metric on the m^{th} layer defined as $\sum_{j \in N(T_\sigma^m) \cap S(i, m+1)} (\deg(j) - 1 - np)$, however other metrics that encode the degree information of this layer would work as well.

One of the key assumptions to this algorithm actually producing a *partition* of the r^{th} layer is that $S(i, r)$ is a tree in A . This is not always the case, but is so on most $(n - n^{1-c})$ w.h.p of $[n]$. The details and difficulty of the analysis is mostly due to this, so I will just sweep it under the rug and assume every vertex has a tree in its m -neighborhood, namely that $S(i, m)$ has no cycles for all $i \in [n]$ for $m = C \log \log n$. WLOG let the latent permutation π^* be the identity for ease of notation.

Given this we can say i has

- $|S(i, 1)| \approx np \implies |T_\sigma^1| \approx \frac{np}{2}$. Then the noise comes in for $|T_\sigma^1(i, A) \cap T_\sigma^1(i, B)| \approx \frac{np}{2}(1 - \kappa(\delta))$ for some function κ that goes to 0 as $\delta \rightarrow 0$.
- Repeating this argument we see $|S(i, m)| \approx (np)^m \implies |T_\sigma^m| \approx (\frac{np}{2})^m$ and the noise causes a similar $|T_\sigma^m(i, A) \cap T_\sigma^m(i, B)| \approx (\frac{np}{2})^m (1 - \kappa(\delta))^m$.

Thus i in A and i in B have a large overlap and therefore similar signatures, while the overlap of i and $j \neq i$ will be small:

$$\frac{(f_i^A - f_i^B)_\sigma^2}{\text{variance}} \leq 1 - (1 - 2\kappa(\delta))^m \leq \left(1 - \frac{1}{c'}\right)^{C \log \log n} \leq 1 - (\log n)^{C/c'} \text{ for some appropriate constant } c' \quad (1)$$

$$\frac{(f_i^A - f_j^B)_\sigma^2}{\text{variance}} \approx 1. \quad (2)$$

With this we've almost created a good matching. The only issue remaining is that these signatures are not independent. To get around this we use sparsification.

4.2 Sparsification

To avoid the dependence of the partition trees of nearby vertices we take a uniformly random subset $I \subset \{\pm 1\}^m$ with $|I| = \text{polylog}(n) \ll 2^m$ of the leaves when comparing the signatures. Then we match i and j if and only if

$$\frac{1}{|I|} \sum_{\sigma \in I} \frac{(f_i^A - f_i^B)^2}{\text{variance}} \leq 1 - \frac{1}{\sqrt{\log n}}.$$

The $\frac{1}{\sqrt{\log n}}$ is just a specific value for our $(\log n)^{C/c'}$ in Equation (1). This process will then produce an almost exact matching $\hat{\pi}$ with $|\{i : \hat{\pi}(i) \neq \pi^*(i)\}| \leq 4n^{1-c}$ with probability $1 - n^{-10}$ (coming from proportion of “good” vertices).

4.3 Exact matching

We use an iterative procedure to prove Theorem B defined in Algorithm 2. This would work for any $\hat{\pi}$ with $|\{i : \hat{\pi}(i) \neq \pi^*(i)\}| \leq \lambda n$, a weaker condition than what we have from Theorem A. The algorithm, on each iteration, takes only those vertices we can be sure of being the same using their local neighborhoods, and then just extends the permutation arbitrarily.

Algorithm 2 RefinedMatching

Input: Two graphs A and B on $[n]$, a permutation $\hat{\pi} : [n] \rightarrow [n]$, and a parameter $\varepsilon > 0$

Output: A permutation $\tilde{\pi} : [n] \rightarrow [n]$

```

1:  $\pi_0 \leftarrow \hat{\pi}$ 
2: for  $\ell = 1, \dots, \lceil \log_2 n \rceil$  do
3:   for  $i = 1, \dots, n$  do
4:     if there is a vertex  $i' \in [n]$  such that
5:       •  $|\pi_{\ell-1}^{-1}(N_A(i)) \cap N_B(i')| \geq \varepsilon^2 pn$ 
6:       •  $|\pi_{\ell-1}^{-1}(N_A(i)) \cap N_B(j')| < \varepsilon^2 pn$  for all  $j' \in [n] \setminus \{i'\}$ 
7:       •  $|\pi_{\ell-1}^{-1}(N_A(j)) \cap N_B(i')| < \varepsilon^2 pn$  for all  $j \in [n] \setminus \{i\}$ 
8:     then
9:        $\pi_\ell(i') \leftarrow i$ 
10:    end if
11:  end for
12:  extend  $\pi_\ell$  to a permutation on  $[n]$  in an arbitrary way
13: end for
14:  $\tilde{\pi} \leftarrow \pi_{\lceil \log_2 n \rceil}$ 
15: return  $\tilde{\pi}$ 

```

This algorithm is correct because in each iteration, the partial matching π_ℓ is an improvement over $\pi_{\ell-1}$ with high probability. For example the first step has $|\{i : \pi_1(i) \neq \pi^*(i)\}| \leq \lambda n/2$ with probability $1 - n^{-\varepsilon/10}$, which means the next iteration will have $|\{i : \pi_\ell(i) \neq \pi^*(i)\}| \leq \lambda n/2^\ell$ with high probability (see section 7 of [MRT] for more details). This means $\pi_{\lceil \log_2 n \rceil} = \pi^*$ is an exact matching. Also, the algorithm runs in $O(n^2)$ for each iteration, and therefore terminates in $O(n^{2+o(1)})$ producing an exact matching.

References

- [MRT] Cheng Mao and Mark Rudelson and Konstantin Tikhomirov. Exact Matching of Random Graphs with Constant Correlation. *arXiv preprint arXiv:2110.05000*, 2021.
- [Bab16] László Babai. Graph isomorphism in quasipolynomial time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 684–697, 2016.
- [BCL⁺19] Boaz Barak, Chi-Ning Chou, Zhixian Lei, Tselil Schramm, and Yueqi Sheng. (nearly) efficient algorithms for the graph matching problem on correlated random graphs. In *Advances in Neural Information Processing Systems*, pages 9190–9198, 2019.
- [BCPP98] Rainer E Burkard, Eranda Cela, Panos M Pardalos, and Leonidas S Pitsoulis. The quadratic assignment problem. In *Handbook of combinatorial optimization*, pages 1713–1809. Springer, 1998.
- [DMWX21] 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.
- [FMWX19] 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*, 2019.
- [GML21] Luca Ganassali, Laurent Massoulié, and Marc Lelarge. Correlation detection in trees for partial graph alignment. *arXiv preprint arXiv:2107.07623*, 2021.
- [MRT21] Cheng Mao, Mark Rudelson, and Konstantin Tikhomirov. Random graph matching with improved noise robustness. In *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 3296–3329. PMLR, 15–19 Aug 2021.
- [MX20] Elchanan Mossel and Jiaming Xu. Seeded graph matching via large neighborhood statistics. *Random Structures & Algorithms*, 57(3):570–611, 2020.
- [PRW94] Panos M. Pardalos, Franz Rendl, and Henry Wolkowicz. The quadratic assignment problem: A survey and recent developments. In *In Proceedings of the DIMACS Workshop on Quadratic Assignment Problems, volume 16 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 1–42. American Mathematical Society, 1994.
- [WXY21] Yihong Wu, Jiaming Xu, and Sophie H. Yu. Settling the sharp reconstruction thresholds of random graph matching. *arXiv preprint 2102.00082*, 2021.