

Spectral Graph Matching and Quadratic Relaxations

Noora Torpo & Titouan Renard

May 18, 2022

1 The Graph Matching Problem

In the following section we define the graph matching problem and then we discuss the Gaussian-Wigner model of graph matching.

1.1 General problem definition and initial remarks

Problem 1.1. Given two edge-weighted graphs G_A, G_B , s.t. $|V(G_A)| = |V(G_B)|$ and $|E(G_A)| = |E(G_B)|$, let A and B denote their respective weighted adjacency matrices. **Graph Matching** (or network alignment) refers to finding a bijection π^* (which we can think of as a permutation $\pi \in \mathcal{S}_n$ on the n labels of the vertices in the set $V(G_B)$) between the vertex sets V_A, V_B of so that their edge sets E_A, E_B are maximally aligned (with respect to their weights). The optimal solution π^* of the graph matching problem satisfies:

$$\max_{\pi^* \in \mathcal{S}_n} \sum_{i,j=1}^n A_{ij} B_{\pi(i)\pi(j)}. \quad (1)$$

Definition 1.1. (*Adjacency Matrix*) *Properties of the weighted adjacency matrix*, recall that given a graph $G = (V, E)$ and an edge-weighting function $w : E \rightarrow \mathbb{R}^+$ the adjacency matrix A associated with G, w is constructed as:

$$(A_{ij} = w(e[i, j])).$$

Definition 1.2. (Spectral Decomposition) Given the adjacency matrix A of the graph G we write it's spectral decomposition as:

$$(A_i j) = \left(\sum_{ij}^n \lambda_i u_i u_i^T \right),$$

$$A = Q \Lambda Q^T.$$

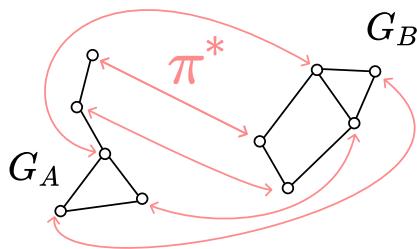


Figure 1: A maximal graph matching for two graphs.

1.2 The Gaussian Wigner Model

Definition 1.3. (*Gaussian Orthogonal Ensemble*) we say that $A \in \mathbb{R}^{n \times n}$ is from the Gaussian Orthogonal Ensemble ($A \sim GOE(n)$) if A is symmetric and

$$(A_{ij}) = \begin{cases} \sim \mathcal{N}(0, \frac{1}{n}) & i \neq j, \\ \sim \mathcal{N}(0, \frac{2}{n}) & i = j. \end{cases}$$

Definition 1.4. (*Gaussian Wigner Model*) We say that the graph pair G_A, G_B described by the matrix pair $A, B \in \mathbb{R}^{n \times n}$ follows the Gaussian Wigner model if:

$$B^{\pi^*} = A + \sigma Z, \quad (2)$$

where $A, Z \sim GOE(n)$ are independent and $\sigma \geq 0$ and π^* is the permutation that we try to recover.

2 A spectral algorithm for the graph matching problem

In the following section we use the following notation for the spectral decomposition of the adjacency matrices A and B of graphs G_A and G_B as:

$$A = \sum_{i=1}^n \lambda_i u_i u_i^T, \quad B = \sum_{j=1}^n \mu_j v_j v_j^T,$$

where the eigenvalues are sorted such that:

$$\lambda_1 \geq \dots \geq \lambda_n, \quad \mu_1 \geq \dots \geq \mu_n.$$

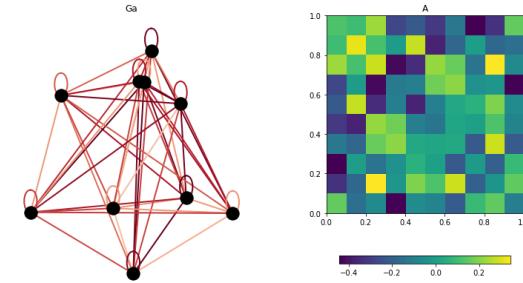


Figure 2: $GOE(5)$ complete graph G_a (left) and it's adjacency matrix A (right).

Property 2.1. (equivariance) let

$\hat{\pi}(A, B)$ denote the output of Algorithm

1, for any given permutation π on B (which gives the permuted matrix B^π constructed as $B_{ij}^\pi = B_{\pi i, \pi j}$) we have the following property:

$$\pi \circ \hat{\pi}(A, B^\pi) = \hat{\pi}(A, B) \quad (6)$$

Algorithm 1: Graph Matching by Pairwise eigen-Alignment (GRAMPA)

Input: Weighted adjacency matrices $A, B \in \mathbb{R}^{n \times n}$ and tuning parameter $\eta \in \mathbb{R}$

Output: A permutation $\hat{\pi} \in \mathcal{S}_n$
Construct the similarity matrix:

$$\hat{X} = \sum_{i,j}^n w(\lambda j, \mu_j) u_i^T u_i \mathbf{J} v_j v_j^T \in \mathbb{R}^{n \times n}, \quad (3)$$

where \mathbf{J} denotes the all ones $\mathbb{R}^{n \times n}$ matrix and w is the Cauchy kernel of bandwidth η .

$$w(x, y) = \frac{1}{(x - y)^2 + \eta^2}. \quad (4)$$

Output the permutation estimate $\hat{\pi}$ by rounding \hat{X} to a permutation by solving the following linear assignment problem:

$$\hat{\pi} = \underset{\pi \in \mathcal{S}_n}{\operatorname{argmax}} \sum_{i=1}^n \hat{X}_{i,\pi(i)}. \quad (5)$$

2.1 Theorem 2.1 Outline

The above described algorithm relays firmly on a theorem presented in the paper. At the last step of the algorithm, the similarity matrix is rounded to a vector that is claimed to present the estimation of the search permutation (add ref to line in algo). As we can see rounding is done by choosing the permutation that maximizes the sum of the diagonal elements after permuting the columns of the similarity matrix. The theorem, that ensures that this is indeed the best possible permutation, informally tells us that the diagonal elements of the similarity matrix are most likely higher than the other elements.

Theorem 2.1. *Consider the model (ref). There exist constants $c, c' > 0$ such that if*

$$1/n^{0.1} \leq \eta \leq c/\log n \quad \text{and} \quad \sigma \leq c'\eta,$$

then with probability at least $1 - n^{-4}$ for all large n , the matrix \hat{X} in (ref) satisfies

$$\min_{i \in [n]} \hat{X}_{i,\pi^*(i)} > \max_{i,j \in [n]: j \neq \pi^*(i)} \hat{X}_{i,j} \quad (7)$$

and hence, Algorithm 1 recovers $\hat{\pi} = \pi^$.*

To gain some intuition on why this proofs that the recovered permutation is the optimal one, let us verbalize what this result means. The theorem is claiming that most likely the "permuted diagonal" is always larger than the other elements in the matrix. This means that the similarity to the corresponding node between the two graphs is higher than the similarity between any other node. In noiseless setting this is obvious, the similarity to the node itself is clearly higher than or equal to

the similarity with respect to other nodes. The theorem states that under some condition we can trust that most likely the same is observed even if there is noise.

Let us next have a glance at the proof of the theorem. The property 2.1 (equivariance) plays an important role in it. The property allows us to examine the identity permutation $\pi(i) = i$ because the corresponding result should be found with all permutations. Therefore we can examine two matrices: $X = \hat{X}(A, B)$ which is the similarity matrix between the graphs A and B , and $X_* = \hat{X}(A, A)$ which presents the similarity matrix in a noiseless setting.

The writers present and proof two important lemmas that are used to complete the proof of the theorem. In short, the other lemmas states that under some conditions we gain lower bound for the diagonal elements of X_* and upper bound to the other elements. The second lemma gives an upper bound for the maximum difference between the elements in X_* and X . Note that both lemmas, work in a stochastic setting so the results hold with some probability. Large part of the paper is used to proof these two lemmas, but let us concentrate on the higher level induction. Combining these two lemmas and the equivariance property, the theorem is proved.

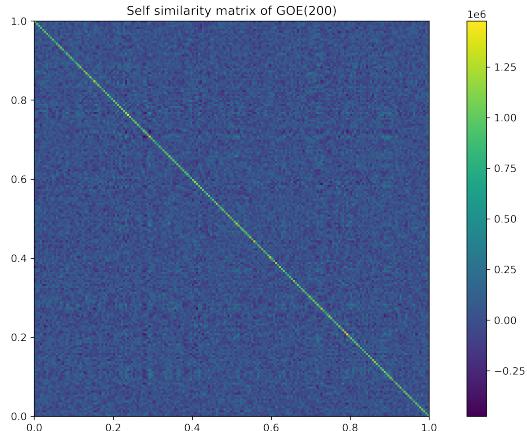


Figure 3: Illustration of the diagonal-dominance proof: self similarity matrix \hat{X} of A drawn from $GOE(200)$.

3 An optimization perspective on the problem

The problem as stated in (1) is an instance of the quadratic assignment problem a (QAP) which is NP-hard. Which can also be written in matrix notation as a combinatorial optimization problem on permutation matrices $\Pi \in \mathcal{G}_n$:

$$\max_{\Pi \in \mathcal{G}_n} \langle A, \Pi B \Pi^T \rangle \iff \min_{\Pi \in \mathcal{G}_n} \|A - \Pi B \Pi^T\|_F^2 \iff \min_{\Pi \in \mathcal{G}_n} \|A\Pi - \Pi B\|_F^2$$

Which can in turn be relaxed from the set of permutations to it's convex hull (*the Birkhoff polytope*) into the following a quadratic program which is convex with respect to the relaxed permutation matrix X :

$$\min_{X \in \mathcal{B}_n} \|AX - XB\|_F^2 \quad (8)$$

Here the intuition is that a valid permutation matrix is some matrix filled with values $\Pi_{ij} \in \{1, 0\}$, s.t all columns and all rows sum to 1, we will find an approximation of the solution by finding an approximate solution matrix X s.t. $X_{ij} \in [0, 1]$ and all column and rows sum to 1, which we will round to a permutation matrix.

The set of all such matrices is given by the **Birkhoff polytope** which we formally define as:

$$\mathcal{B}_n := \{X \in \mathbb{R}^{n \times n} : X\mathbf{1} = \mathbf{1}, X^T\mathbf{1} = \mathbf{1}, X_{ij} \geq 0 \ \forall i, j\},$$

with $\mathbf{1}$ denoting the all ones $1 \times n$ vector.

We can show that the spectral method of Alg. 1 outputs a matrix \hat{X} which is a minimizer of the following quadratic problem:

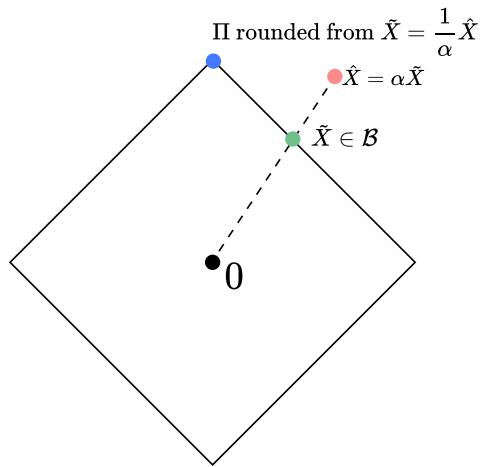
$$\min_{X \in \mathbb{R}^{n \times n}} \|AX - XB\|_F^2 + \frac{\eta^2}{2} \|X\|_F^2 - \mathbf{1}^T X \mathbf{1}. \quad (9)$$

Observe that the solution of the unconstrained \hat{X} of (9) is a scalar multiple of the solution \tilde{X} of the constrained problem below :

$$\begin{aligned} & \min_{X \in \mathbb{R}^{n \times n}} \|AX - XB\|_F^2 + \frac{\eta^2}{2} \|X\|_F^2, \\ & \text{s.t. } \mathbf{1}^T X \mathbf{1} = n. \end{aligned}$$

Where one can think of \tilde{X} as a projection of \hat{X} onto \mathcal{B} (see figure 4). Differentiating the cost of (9) by X yield the gradient $A^2X + XB^2 - 2AXB + \eta^2X - \mathbf{J}$, which naturally gives gradient descent dynamics.

Figure 4: A visual illustration of the quadratic unconstrained relaxation to the assignment problem.



How do we show that?
Gradient descent on (9).

Titou writes out the convergence proof

4 Results

Noora