Optimizing the overlap of two independent Erdős-Rényi graphs

Shuyang Gong

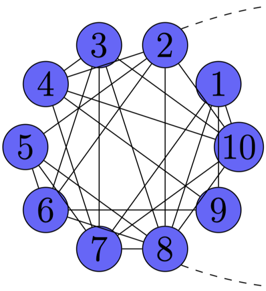
School of Mathematical Sciences, Peking University

Janurary, 2024

Joint work with Jian Ding (PKU), Hang Du (MIT) and Rundong Huang (PKU)

Motivations I: random graph matching

- **Random Graph Matching** is an extensively studied topic in recent years, which lies in the intersection of **probability, statistics and theoretical computer science**.
- Goal: find a bijection between two vertex sets which maximizes the number of common edges (i.e. minimize the adjacency disagreements)



QAP

- Given symmetric $n \times n$ matrices A and B, solve **Quadratic Assignment Problem(QAP)**:

$$\max_{\pi \in S_n} \sum_{i<j} A_{i,j} B_{\pi(i),\pi(j)}.$$

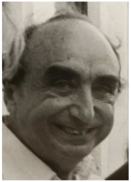
- Introduced by [Koopmans-Beckmann Econometrica'57] .

COWLES FOUNDATION DISCUSSION PAPER, NO. 4*

Assignment Problems and the Location of Economic Activities**

by

Tjalling C. Koopmans and Martin Beckmann

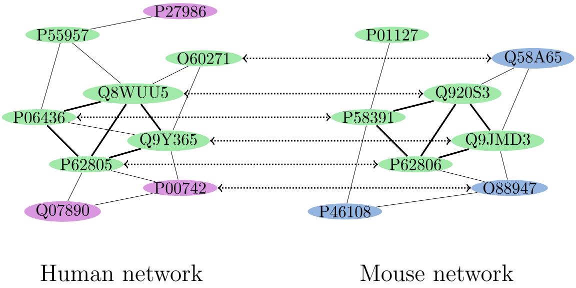
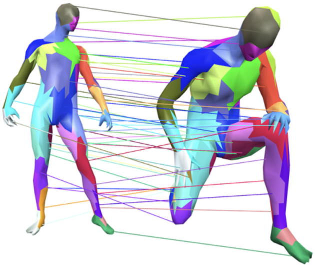


Figure: Protein-protein interaction

Graph matching for aligning protein-to-protein interaction networks between two species, to identify conserved components and genes with common function. [Singh-Xu-berger'08]

Application 2: computer vision

A fundamental problem in computer vision: Detect and match similar objects that under go different deformations



3D shapes $\rightarrow$ geometric graphs (features $\rightarrow$ nodes, distances $\rightarrow$ edges)

Key challenges

- Statistical: two graphs are usually not isomorphic.
- Computational: $n!$ bijections. ($100! \approx 10^{158}$)
- NP-hard in the worst case: QAP is hard to approximate within $\exp(\mathrm{polylog}(n))$ mutiplicative factor. [Makarychev-Manokaran-Sviridenko '15]
- Efficient algorithms for average-case analysis are expected.
- Efforts from community on **average-case** of random graph matching: [Feizi at el.'16, Lyzinski at el'16, Cullina-Kiyavash'16,17, Ding-Ma-Wu-Xu'18, Barak-Chou-Lei-Schramm-Sheng'19, Fan-Mao-Wu-Xu'19a,19b, Ganassali-Massoulié'20, Hall-Massoulié'20, Ding-Du'22a,22b, Ding-Du-G'22, Ding-Li'22,23, Du-G-Huang'23, Ding-Du-Li'23...]
- It is fair to say that there is still a long way to go to understand the real networks. But it is the first step forward...

Motivation II: random optimization problem

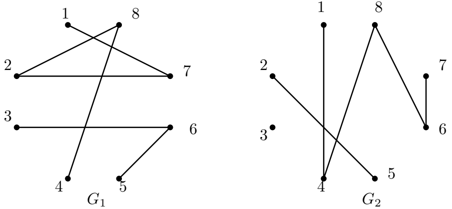
- **Random optimization problem** refers to solving optimization problems where the instance is randomly sampled.
- The problems arise from various fields including **computer science, statistical physics, operations research**. (e.g. finding maximal independent set in a random graph [Rahman-Virág'17,Wein'22] / finding the groundstate energy of Hamiltonian in spin glass models [Huang-Sellke'22]...)
- **Key challenge**: non-convexity and high-dimensionality. (case by case analysis)
- **Central Question**: efficient algorithm exists? information-computation gap?
- Efforts from the community on random optimization problems: [Ding-Du-G'22, Du-G-Huang'23, Gamarnik'21, Garmarnik-Moore-Zdeborová'22, Garmarnik-Kızıldağ-Perkins-Xu'23, Gamarnik-Sudan'14, Garmarnik-Zadik'19, Huang-Sellke'22, Montanari'19, Rahman-Virág'17, Subag'21, Wein'22...]

Mathematical model

- Erdős-Rényi graph $G(n,p)$: Each edge in K_n is preserved with probability p independently.
- Sample two independent Erdős-Rényi graphs $G_1(n,p)$ and $G_2(n,p)$.
- **Core quantity** $O(\pi)$: the number of **common edges** of these two graphs under π. Formally,

$$O(\pi) := \sum_{i<j} G^{(1)}_{i,j} G^{(2)}_{\pi(i),\pi(j)} \,,$$

  where $G^{(i)}$ are adjacency matrices.
- e.g. $\pi(1)=1, \pi(2)=8, \pi(3)=2, \pi(4)=7, \pi(5)=3, \pi(6)=5, \pi(7)=4, \pi(8)=6 \Rightarrow$ we have $O(\pi)=5$



Our problem

- **Q1: what is the typical value of** $\max_{\pi \in S_n} O(\pi)$?
- A first moment computation on $\max_{\pi \in S_n} O(\pi)$ yields an **upper bound**, e.g. take $p = n^{-3/4}$, let $\gamma(n) := (1+\varepsilon)2n$,

$$
\begin{aligned}
&\mathbb{P}\left[\max_{\pi \in S_n} O(\pi) > 2(1+\varepsilon)n\right] \leq \sum_{\pi \in S_n} \mathbb{P}\left[O(\pi) \geq 2(1+\varepsilon)n\right] \\
=& n!\mathbb{P}\left[\mathbf{B}\left(\binom{n}{2}, p^2\right) > 2(1+\varepsilon)n\right] \\
\overset{\text{Chernoff}}{\leq}& n! \exp\left(-2(1+\varepsilon)n \log\left(\frac{2(1+\varepsilon)n}{\binom{n}{2} n^{-3/2}}\right) + 2(1+\varepsilon)n - \binom{n}{2} n^{-3/2}\right) \\
=& n! \exp(-(1+\varepsilon+o(1))n \log n) = o(1)\,.
\end{aligned}
$$

- The calculation for other p is similar.

Our problem

- For other p (divide into sparse/dense by $\sqrt{\log n / n}$),

regime	$\max_{\pi \in S_n} O(\pi)$
sparse: $\frac{\log n}{n} \ll p \ll \sqrt{\frac{\log n}{n}}$	$n \cdot \frac{\log n}{\log(\log n / np^2)}$
dense: $\sqrt{\frac{\log n}{n}} \ll p \leq \frac{1}{(\log n)^4}$	$\binom{n}{2} p^2 + \sqrt{n^3 p^2 \log n}$

- First moment computation $\Rightarrow$ Upper bound w.h.p.
- Right asymptotics?—True.
- **Q2: Find a polynomial time algorithm for** $\arg\max_\pi O(\pi)$**?** (sparse: yes, dense: no)
- **Information-Computation gap?** (sparse: no, dense: yes)

Sparse regime

Theorem (Ding-Du-G. 22)

For $p = n^{-\alpha+o(1)}$, $1/2 < \alpha \le 1$, there exists a polynomial-time algorithm s.t.

$$\mathbb{P}\left[\mathrm{O}(\pi^*) \ge \frac{1-\epsilon}{2\alpha-1} n\right] = 1 - o(1).$$

Theorem (Du-G.-Huang 23)

For $p = n^{-1/2+o(1)}$ and $p \ll \sqrt{\log n/n}$, for any $\varepsilon > 0$, there exists an $O(n^3)$-time algorithm such that

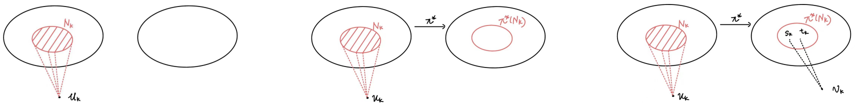
$$\mathbb{P}\left[\mathrm{O}(\pi^*) \ge \frac{(1-\varepsilon) n \log n}{\log\left(\log n/np^2\right)}\right] = 1 - o(1)\,.$$

- $n/(2\alpha - 1) = (1 + o(1)) n \log n / \log\left(\log n/np^2\right)$ for $p = n^{-\alpha}$.
- The **constructive lower bound** matches the $\gamma(n)$ derived in the first moment computation.
- **No information-computation gap** in the sparse regime.

Algorithm

The **greedy** algorithm in [Ding-Du-G'22], let $\alpha = 3/4 - \delta$, $\frac{1}{2\alpha - 1} n \approx 2n$

- Match the first εn vertices arbitrarily.
- In step $k+1$, select unmatched u_k in G_1. Neighbor of u_k in matched part N_k
- Map N_k by π^*
- For each such pair (s_k, t_k), check if there exists unmatched v_k in G_2
- If succeed, let $\pi^*(u_k) = v_k$.
- For other α, match a carefully designed tree in each step.



(i) (ii) (iii)

The idea: why the algorithm works?

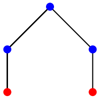
- Consider the case $\alpha = 3/4 - \delta$. Assume independence among the iterative steps.
- $N_k \sim \mathbf{B}(n, p)$.
- Conditioned on N_k. For each v_k in G_2, the number of edges s.t. $v_k \to \pi^*(N_k)$ obeys $\mathbf{B}(|N_k|, p)$.
- By Poisson approximation,

$$\mathbb{P}[\mathbf{B}(|N_k|, p) \geq 2] = \theta((np^2)^2) = \theta(n^2 p^4)\,.$$

- All v_k fail with probability $(1 - n^2 p^4)^n \sim \exp(-n^3 p^4) \to 0$.
- It suffices to tackle the **dependence**.

Key technical input: dealing with correlations

- Consider the (slightly more complicated) case $\alpha = 7/8 - \delta$, $1/(2\alpha - 1) = 4/3$.



- To show
$$\mathbb{P}[L \sim Q \mid \mathcal{A}_1, \mathcal{A}_2] = (1 + o(1))\mathbb{P}[L \sim Q]\,.$$
- $\mathcal{A}_1$ and $\mathcal{A}_2$ is the set of "failure" and "successful" trees.
- LHS equals to
$$\frac{\mathbb{P}[L \sim Q, \mathcal{A}_1 \mid \mathcal{A}_2]}{\mathbb{P}[\mathcal{A}_1 \mid \mathcal{A}_2]} = \frac{\widehat{\mathbb{P}}[\mathcal{A}_1 \mid L \sim Q]}{\widehat{\mathbb{P}}[\mathcal{A}_1]}\mathbb{P}[L \sim Q]\,.$$
- Equivalently, to show the first factor is $1 - o(1)$.
- It means that if we open the edges in $L \sim Q$, at least one of the "failure" trees emerges.

Key technical input: counting intersection patterns

- The possible intersection patterns:

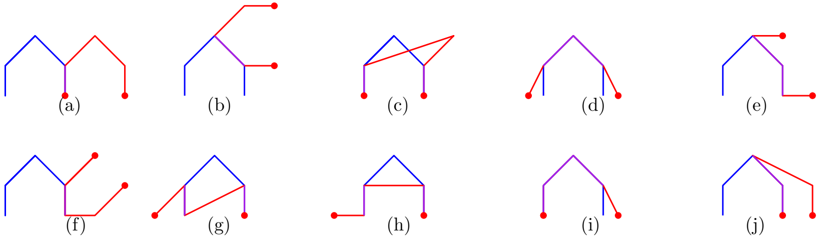


Figure: Intersection patterns

the blue tree represents $L \sim Q$, the red one is from $\mathcal{A}_1$.

- (Union bound) count the total number of such intersections:

$$\text{number of leaves} \times \text{number of non-leaves} = o(1)\,.$$

Dense regime-information

Theorem (Du-G.-Huang 23, informational result)

For p in the dense regime, we have

$$\frac{\max_{\pi \in \mathsf{S}_n} \mathsf{O}(\pi) - \binom{n}{2} p^2}{\sqrt{n^3 p^2 \log n}} \overset{prob.}{\to} 1 \,.$$

- **Second moment method:** $X_\varepsilon := \sum_{\pi \in \mathsf{S}_n} \mathbf{1}_{\mathsf{O}(\pi) > \binom{n}{2} p^2 + \sqrt{(1-\varepsilon) n^3 p^2 \log n}}$.

$$\mathbb{P}\left[\max_{\pi \in \mathsf{S}_n} \mathsf{O}(\pi) > \binom{n}{2} p^2 + \sqrt{(1-\varepsilon) n^3 p^2 \log n} \right] \geq \frac{(\mathbb{E} X_\varepsilon)^2}{\mathbb{E} X_\varepsilon^2} = \exp(-o(n \log n)) \,.$$

- **Idea:** concentration of maximum.
- **Talagrand's concentration inequality:**

$$\mathbb{P}\left[\left| \max_{\pi \in \mathsf{S}_n} \mathsf{O}(\pi) - \mathbb{E} \max_{\pi \in \mathsf{S}_n} \mathsf{O}(\pi) \right| \geq \sqrt{\varepsilon n^3 p^2 \log n} \right] \leq \exp\left(-c(\varepsilon) n \log n \right) \,.$$

Dense regime-computation

Theorem (Du-G.-Huang 23, computational result)

There exists an $O(n^3)$-time algorithm $\mathcal{A}$ which outputs a π^* such that

$$\mathbb{P}\left[\frac{\mathsf{O}(\pi^*) - \binom{n}{2}p^2}{\sqrt{n^3 p^2 \log n}} \geq \sqrt{\mathbf{8/9}} - \varepsilon\right] = 1 - o(1)\,.$$

Theorem (Du-G.-Huang 23, hardness result)

For p in the dense regime, for all $\varepsilon > 0$, there exists a constant $c = c(\varepsilon) > 0$ such that for any online algorithm $\mathcal{A}$,

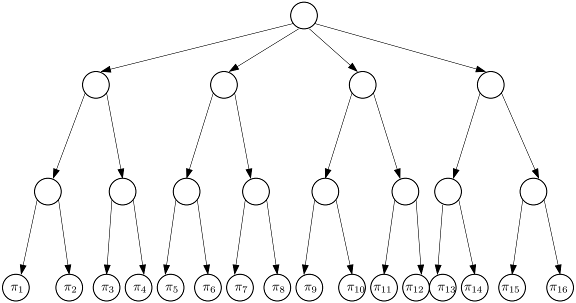
$$\mathbb{P}\left[\frac{\mathsf{O}(\mathcal{A}(\mathsf{G}_1, \mathsf{G}_2)) - \binom{n}{2}p^2}{\sqrt{n^3 p^2 \log n}} \geq \sqrt{\mathbf{8/9}} + \varepsilon\right] = \exp(-c(\varepsilon) n \log n)\,.$$

No online algorithm above $\sqrt{8/9}$! — Hardness result.

- Main tool: **Branching-OGP structure**[Huang-Sellke'22].

The Branching OGP

- Define tree $\mathbb{T}$ with leave set $\mathbb{L}$.
- Construct leave-indexed correlated instances $\{(\mathsf{G}_1^{(u)}, \mathsf{G}_2)\}_{u\in\mathbb{L}}$.
- Each ray represents an instance. $G_1^{(u)}$ and $G_1^{(v)}$ share $\rho_{|u\wedge v|}$ Bernoulli variables. $(\rho_1 < \rho_2 < \rho_3)$
- Impossible for all $\mathsf{O}(\pi_i)$ above $\sqrt{8/9} + \varepsilon$.
- Run online algorithm on all instances. Prove by contradiction.



Open problems

- How about $p = \theta(\sqrt{\log n/n})$?
- When $1/2 < \alpha < 1$, is there a polynomial-time algorithm with fixed power that finds (near) optimal matchings?
- For other graph model (with more general edge weights), determine the maximal overlap.

Summary

Take-home messages

- In sparse regime, no information computation gap.
- In dense regime, information computation gap emerges with a threshold $\sqrt{8/9}$.

Related papers:

DDG22 Jian Ding, Hang Du and Shuyang Gong, A Polynomial-time Approximation Scheme for the Maximal Overlap Between Two Independent Erdős-Rényi Graphs. To appear in *Random Structures and Algorithms*.

DGH23 Hang Du, Shuyang Gong and Rundong Huang, The Algorithmic Phase Transition of Random Graph Alignment Problem, *arXiv:2307.06590*.

References

[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.

[DDG22] Jian Ding, Hang Du and Shuyang Gong, A Polynomial-time Approximation Scheme for the Maximal Overlap Between Two Independent Erdős-Rényi Graphs. To appear in *Random Structures and Algorithms*.

[DGH23+]Hang Du, Shuyang Gong and Rundong Huang, The Algorithmic Phase Transition of Random Graph Alignment Problem, *arXiv:2307.06590*.

[HS22] B. Huang and M. Sellke, "Tight Lipschitz Hardness for optimizing Mean Field Spin Glasses," *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS), Denver, CO, USA, 2022, pp. 312-322, doi: 10.1109/FOCS54457.2022.00037*.

[PRW94] Panos M. Pardalos, Franz Rendl, and Henry Wolkowicz. The quadratic assignment problem: A survey and recent developments.