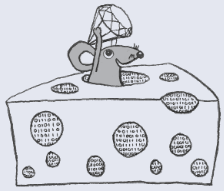


Sublinear Algorithms Open Problem 4

Some open problems on 2-coloring and 3-coloring

Lecturer: Akash Kumar. *Scribe:* Akash Kumar

June 14, 2024

In all the problems listed in this section, we will always assume that graphs in question are bounded degree. That is, $\Delta(G) \leq d = O(1)$. Alright, let's begin.

1 Better than 1/2 sublinear approximation for max-cut

Assume query access to the adjacency list of G (a degree bounded graph), allowing both degree queries and neighboring queries. Fix some sufficiently small $\varepsilon > 0$. Do there exist algorithms that run in sublinear time which decide correctly with probability $> 2/3$ which among the following two alternatives holds.

- Is $MC(G) \geq 1 - \varepsilon$, Or
- Is $MC(G) \leq 1/2 + \varepsilon$.

2 Is the spectrum of a nearly bipartite graph kinda symmetric?

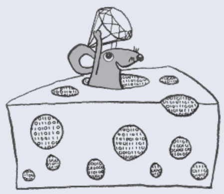
Let $\boldsymbol{L}$ denote the normalized Laplacian of a degree bounded graph with eigenvalues

$$0 = \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \ldots \leq \lambda_n \leq 2.$$

Suppose we know that for some sufficiently small $\varepsilon > 0$, it holds that $\lambda_{n-k+1} \geq 2 - \varepsilon$. Then $\lambda_k \leq \varepsilon \cdot poly(k)$.

3 Lower Bounds for testing 3-colorability

Does testing 3-colorability on a degree bounded graph admit a 2-sided lower bound of $\Omega(n)$ queries even when restricted to distinguishing 3-colorable graphs from graphs that are 1/3-far from being 3-colorable?



Sublinear Algorithms Open Problem 3

Approximate counting of permutation patterns

Lecturer: Omri Ben-Eliezer. *Scribe:*

July 18, 2024

This note concerns pattern counting problems in permutations/sequences. A permutation $\pi\colon [n] \to [n]$ contains a permutation pattern $\sigma\colon [k] \to [k]$ in locations $i_1 < i_2 < \ldots < i_k \in [n]$ if the subsequence of π in these locations has the same "relative order" as in σ. That is, for each $j, j' \in [k]$, we have $\pi(i_j) < \pi(i_{j'})$ if and only if $\sigma(j) < \sigma(j')$.

A breakthrough result of Guillemot and Márx [GM14] shows that *detection* of permutation patterns, i.e., the problem of determining whether π contains at least one copy of σ, is a fixed parameter tractable problem: it requires time $O(n)$ when k is fixed. The dependence they obtained is $2^{O(k^2 \log k)}n$, later slightly improved by Fox [Fox13] to $2^{O(k^2)}n$. The algorithm of Guillemot and Marx played a seminal role in the development of the twin-width, a central notion of modern parameterized complexity which generalizes tree-width (as well as an impressive number of other notions). That said, the story of the dependence in k is not yet complete; could the right complexity be closer to a single exponential in k, i.e., $2^{\tilde{O}(k)}n$?

Exact counting of permutation patterns, i.e., the task of returning the exact number of copies of σ in π, is computationally much harder. Berendsohn, Kozma and Marx [BKM21] proved an $n^{\Omega(k/\log k)}$ for this problem conditioned on the exponential time hypothesis (ETH). In the same paper they also proved the best known upper bound (in the regime where k is small), of $n^{\frac{k}{4}+o(k)}$. In fact, there is a conditional separation between detection and counting already for $k = 4$: Dudek and Gawrychowski [DG20] proved that counting 4-pattern is equivalent (by a bidirectional reduction) to counting 4-cycles in sparse graphs. Thus, conditional super-linear lower bounds for the latter translate to similar bounds for counting 4-patterns, answering a question of Even-Zohar and Leng [EZL21].

This substantial gap between detection and counting leads us to ask the following regarding *approximate counting*, i.e., the task of returning the number of σ-patterns in π to within a multiplicative factor of $1 + \varepsilon$.

What is the running time of approximately counting the number of σ-copies in π, as a function of n and k? Is it linear in n, or close to linear, as in detection? Is it of the form $n^{k^{O(1)}}$, as in the exact case?

In ongoing work with Mitrović and Srivastava, we are able to show that at least for $k = 4$ (and possibly $k = 5$), the complexity of approximate counting is much closer to that of detection: we obtain a $\tilde{O}(n)$ upper bound in this case, which breaks the conditional lower bound derived from 4-cycle counting. But our techniques do not seem to generalize to the general k case. In the general case, any super-linear lower bound would be very interesting, as well as any upper bound of the form $n^{o(k/\log k)}$ which would break the conditional lower bound for exact counting. In fact, even beating the state of the art $n^{\frac{k}{4}+o(k)}$ bound by allowing approximation would be interesting.

References

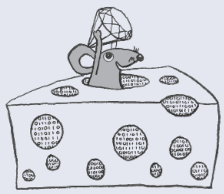
[BKM21] Benjamin Aram Berendsohn, László Kozma, and Dániel Marx. Finding and counting permutations via CSPs. *Algorithmica*, 83(8):2552–2577, 2021.

[DG20] Bartłomiej Dudek and Paweł Gawrychowski. Counting 4-patterns in permutations is equivalent to counting 4-cycles in graphs. In *31st International Symposium on Algorithms and Computation (ISAAC)*, pages 23:1–23:18, 2020.

[EZL21] Chaim Even-Zohar and Calvin Leng. Counting small permutation patterns. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2288–2302, 2021.

[Fox13] Jacob Fox. Stanley-wilf limits are typically exponential, 2013.

[GM14] Sylvain Guillemot and Dániel Marx. Finding small patterns in permutations in linear time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, page 82–101, 2014.



Sublinear Algorithms Open Problem 1

Online Stochastic Matching

Lecturer: Santhoshini Velusamy. *Scribe:* Santhoshini Velusamy

June 7, 2024

1 Introduction

In the online stochastic *hypergraph* matching problem, the edges of a p-partite hypergraph arrive online in a predetermined order. The edges have 0/1 weights which are revealed at the time of arrival. The weights are drawn according to independent Bernoulli distributions (known to the algorithm). The goal is to compute the largest weight maximum matching and each time an edge arrives, the algorithm has to make an *irrevocable* decision to include or exclude the edge from the matching. The approximation ratio of the algorithm is defined as the ratio of expected size of the largest matching to the expected size of the matching output by the algorithm. This problem and several of its variants are well-studied in game theory with well-established connections to auction theory and mechanism design. A simple greedy algorithm achieves p-approximation: greedily include edges with weight 1 as long as the matching constraints are satisfied. Kleinberg and Weinberg [KW12] gave an $\Omega(\sqrt{p})$ lower bound on the approximation ratio. In a joint work with Saxena and Weinberg [SVW23], we recently improved the lower bound to $p^{1/2+\Omega(\frac{1}{\log\log p})}$.

2 Open problems

Question 1: A stronger lower bound. Can we obtain a stronger lower bound of the form $\Omega(p^c)$ for some constant $c > 1/2$?

Question 2: A better upper bound. Is it possible to asymptotically beat the approximation ratio achieved by the greedy algorithm?

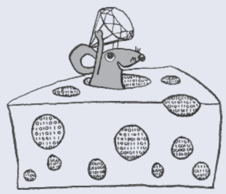
Question 3: Tight bounds for bipartite matching. Even for the special case of bipartite matching, this problem has not been fully resolved. For bipartite matching, the best known algorithm is due to Gravin, Tang, and Wang [GTW21] that achieves 1.988-approximation and the best known lower bound is also due to the same authors who show that no algorithm can achieve better than 3/2-approximation.

References

[GTW21] Nick Gravin, Zhihao Gavin Tang, and Kangning Wang. Online Stochastic Matching with Edge Arrivals. In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, *48th International Colloquium on Automata, Languages, and Programming (ICALP 2021)*, volume 198 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 74:1–74:20, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[KW12] Robert Kleinberg and Seth Matthew Weinberg. Matroid prophet inequalities. In *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '12, page 123–136, New York, NY, USA, 2012. Association for Computing Machinery.

[SVW23] Raghuvansh R. Saxena, Santhoshini Velusamy, and S. Matthew Weinberg. An improved lower bound for matroid intersection prophet inequalities. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*, volume 251 of *LIPIcs*, pages 95:1–95:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.



Sublinear Algorithms Open Problem 2b

Sublinear Time Max Cut in Planar Graphs

Lecturer: Pan Peng. *Scribe:*

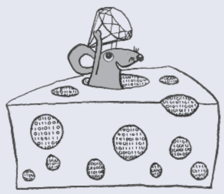
June 14, 2024

1 The Problem

Let $G = (V, E)$ be an unweighted planar graph and let $MC(G) \in [\frac{1}{2}, 1]$ be the maximum fraction of the edges cut by a bipartition of V. Let $\varepsilon > 0$ be any sufficiently small constant.

Given query access to the adjacency list of G, allowing both degree queries and neighboring queries, can we approximate $MC(G)$ within a factor of $1 - \varepsilon$ using $\mathrm{poly}(\log n)$ or even $O_\varepsilon(1)$ queries?

Remark: For bounded-degree graphs, i.e., graphs with maximum degree upper bounded by some constant d, this problem can be solved in $O_{\varepsilon,d}(1)$ queries by making use of the bounded-degree planar partition oracle. So the question here is mainly about *general* planar graph with arbitrary maximum degree.



Sublinear Algorithms Open Problem 2a

Distance Between Vertices after Contraction

Lecturer: Yu Chen. *Scribe:*

June 14, 2024

1 The Problem

Consider a regular random graph G whose degree is some constant, say 100. The diameter of the graph is $\Theta(\log n)$. Suppose we randomly partition vertices into n/s groups, and each group contains s vertices. Then, we contract each group into a super vertex, and with high probability, the new graph has diameter $\Theta(\log n/\log s)$.

The question is, if the partition is not random, can we still say that the distance between vertices is going down to $O(\log n/\log s)$? Note that in this case, the diameter may still be $O(\log n)$. What we can do is the following: we select arbitrary s vertices and compute $\log n/10000$ layers of the BFS tree of all of them. Now we get s same tree. We make the vertices in the same position in different trees as the groups, then after contraction, these super nodes will form the same tree, and thus the diameter is at least $\log n/10000$.

However, the following still might be true: the distance between most of (a large constant fraction) vertices is $O(\log n/\log s)$. So the question is this: is it true that, no matter how we partition the vertices, there is a set of a large fraction of the vertices such that the induced graph has diameter $O(\log n/\log s)$? If we can prove this, then we might generalize it into a hard instance to prove the lower bound on the flow sparsifier problem. If we can disprove this, then we might generalize the partition for any graph and prove the upper bound for the flow sparsifier problem.

References