

Brief Announcement: Congested Clique Counting for Local Gibbs Distributions

Joshua Z. Sobel 

Department of Computer Science, University of Iowa, Iowa City, IA, USA

Abstract

There are well established reductions between combinatorial sampling and counting problems (Jerrum, Valiant, Vazirani TCS 1986). Building off of a very recent parallel algorithm utilizing this connection (Liu, Yin, Zhang arxiv 2024), we demonstrate the first approximate counting algorithm in the CONGESTEDCLIQUE for a wide range of problems. Most interestingly, we present an algorithm for approximating the number of q -colorings of a graph within ϵ -multiplicative error, when $q > \alpha\Delta$ for any constant $\alpha > 2$, in $\tilde{O}\left(\frac{n^{1/3}}{\epsilon^2}\right)$ rounds. More generally, we achieve a runtime of $\tilde{O}\left(\frac{n^{1/3}}{\epsilon^2}\right)$ rounds for approximating the partition function of Gibbs distributions defined over graphs when simple locality and fast mixing conditions hold. Gibbs distributions are widely used in fields such as machine learning and statistical physics. We obtain our result by providing an algorithm to draw n random samples from a distributed Markov chain in parallel, using similar ideas to triangle counting (Dolev, Lenzen, Peled DISC 2012) and semiring matrix multiplication (Censor-Hillel, Kaski, Korhonen, Lenzen, Paz, Suomela PODC 2015). Aside from counting problems, this result may be interesting for other applications requiring a large number of samples.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Distributed algorithms; Theory of computation $\rightarrow$ Random walks and Markov chains

Keywords and phrases Distributed Sampling, Approximate Counting, Markov Chains, Gibbs Distributions

Digital Object Identifier 10.4230/LIPIcs.DISC.2025.65

Related Version *Full Version*: <https://arxiv.org/abs/2508.13083>

Funding The author was partially supported during this project by National Science Foundation grants 1955939 and 2402835.

Acknowledgements Thanks to Sriram Pemmaraju for suggestions and discussions.

1 Introduction

In a pioneering work, Valiant proposed the computational complexity class $\#P$ [12]. Problems in $\#P$ are the counting variants of decision problems in NP; for example, counting the number of proper vertex colorings of a graph using at most q colors (q -colorings). Unfortunately, Valiant established that even when a decision problem is in P , such as determining whether a bipartite graph has a perfect matching, the corresponding counting problem can be NP-hard. Thus for most counting problems, we need to settle for ϵ multiplicative approximation rather than exact counting. The goal is to estimate a true count X by $\hat{X}$, such that $(1 - \epsilon)X \leq \hat{X} \leq (1 + \epsilon)X$ with probability at least $\frac{3}{4}$. By the standard median trick, $3/4$ can be boosted to any constant less than one.

The typical method for approximate counting involves drawing random samples that are solutions to the corresponding search problem. In other words, we can count the number of colorings of a graph by sampling random colorings. This approach dates back to Jerrum, Valiant, and Vazirani. The initial reduction from counting to sampling [8] requires drawing a very large number of samples; this has been improved over several subsequent works [7, 1, 13, 10]. In particular, when considering Gibbs distributions defined over graphs with n



© Joshua Z. Sobel;

licensed under Creative Commons License CC-BY 4.0

39th International Symposium on Distributed Computing (DISC 2025).

Editor: Dariusz R. Kowalski; Article No. 65; pp. 65:1–65:7

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

vertices and meeting mild conditions, Štefankovič, Vempala, and Vigoda [13] showed that counting can be accomplished by taking $\tilde{O}(\frac{n}{\epsilon^2})$ samples. Gibbs distributions originated in physics, and generalize many problems such as the uniform distribution of q -colorings.

Unfortunately, the algorithm from [13] cannot be fully parallelized, as samples drawn later depend on samples drawn earlier. Very recently, Liu, Yin, and Zhang [10] also showed that for many Gibbs distributions defined over graphs counting can be done using $\tilde{O}(\frac{n}{\epsilon^2})$ total samples, but in their approach every sample can be taken in parallel. This implies that counting in the CONGESTEDCLIQUE reduces to quickly drawing $\tilde{O}(\frac{n}{\epsilon^2})$ samples. From existing work, [5, 6, 4], we know that for many Gibbs distributions drawing a single sample within total variation distance error of δ can be done in $O(\log \frac{n}{\delta})$ CONGESTEDCLIQUE rounds, using the *distributed Metropolis-Hastings* Markov chain. For Gibbs distributions that we call *local*, we identify a novel structural similarity between simulating one transition of n instances of the Markov chain and semiring matrix multiplication. In particular, utilizing the approach of Dolev, Lenzen, and Peled [3] for triangle counting and Censor-Hillel, Kaski, Korhonen, Lenzen, Paz, and Suomela [2] for semiring matrix multiplication, we can simulate one transition of n instances of the Markov chain in $O(n^{1/3})$ rounds and thus often obtain an approximate count in $\tilde{O}(\frac{n^{1/3}}{\epsilon^2})$ CONGESTEDCLIQUE rounds. The full version of our paper contains all of our results and proofs.

1.1 Main Results

Our central contribution is an algorithm to draw many samples at once from the distributed Metropolis-Hastings Markov chain. Beyond counting, this result may be useful in other algorithms where several samples are needed.

► **Theorem 1.** *One transition can be simulated for n instances of the distributed Metropolis-Hastings chain in $O(n^{1/3})$ rounds.*

As a corollary to the previous theorem and the results of [10], under mild locality and mixing conditions which we will describe later, there is an $\tilde{O}(\frac{n^{1/3}}{\epsilon^2})$ round counting algorithm. As the most interesting example, we obtain a fast algorithm for approximating the number of q -colorings of an input graph with degree Δ . In addition, our full paper demonstrates a much faster algorithm for the hardcore model with low fugacity.

► **Corollary 2.** *There is an algorithm for approximating the number of q -colorings of a graph in $\tilde{O}(\frac{n^{1/3}}{\epsilon^2})$ rounds, when $q > \alpha\Delta$, for $\alpha > 2$, within multiplicative error ϵ , with probability at least $\frac{3}{4}$.*

We also give a conditional lower bound for approximating a partition function, Theorem 10. Even under our locality and mixing assumptions, counting when $\epsilon \leq \frac{1}{32n^2}$ is as hard as triangle detection.

2 Technical Preliminaries

2.1 Gibbs Distributions on Graphical Models

Consider a traditional graph labeling problem, where each vertex receives a label from some underlying alphabet $\mathcal{A}$. We focus on randomly sampling from a subset, $\mathcal{S}$, of such labelings, given a distribution. In particular, we want to sample from a Gibbs distribution. Borrowing terminology from physics, we have the Hamiltonian, a function $H : \mathcal{S} \rightarrow \{0, 1, \dots, h\}$, for some integer $h \geq 0$.

► **Definition 3** (Gibbs Distribution). *The Gibbs distribution at inverse temperature $\beta \geq 0$ assigns a labeling, $\sigma \in \mathcal{S}$, a probability proportional to $\exp(-\beta H(\sigma))$.*

In the trivial case, if we choose $H = 0$, we get a uniform distribution over $\mathcal{S}$.

► **Definition 4** (Partition Function). *The normalizing constant of the distribution, $Z(\beta) = \sum_{\sigma \in \mathcal{S}} \exp(-\beta H(\sigma))$, is referred to as the partition function.*

We will make the assumption that $|\mathcal{A}|$ and h are bounded by a polynomial in n . This is not a significant restriction and is necessary both algorithmically and to ensure messages are of size $O(\log n)$.

For this paper we will focus heavily on the well studied Potts model.

► **Definition 5.** *The Potts model is a Gibbs distribution where $\mathcal{A} = [q]$, is a set of colors. $\mathcal{S}$ is $[q]^V$. The Hamiltonian, H , assigns to each, possibly improper, coloring the number of monochromatic edges (edges where both vertices have the same color).*

Note that when $\beta = \infty$, the Potts model simply becomes the uniform distribution over proper colorings.¹ Furthermore, $Z(\infty)$ counts the number of proper colorings. On the other hand, when $\beta = 0$, the model is uniform over all proper and improper colorings. Then $Z(0) = q^n$. More precisely, when $\beta > 0$, this model is called the antiferromagnetic Potts model, since neighboring vertices favor not having the same color.

2.2 Counting From Sampling

We would now like to estimate $Z(\beta)$, for a given choice of β . We call this the generalized counting problem, as Z gives the weighted count over all labelings. As discussed in the introduction, it is often sufficient to take $\tilde{O}(\frac{n}{\epsilon^2})$ samples. More precisely, the result from Liu, Yin, and Zhang [10] is summarized by the next theorem.

► **Theorem 6.** *For a Gibbs distribution and inverse temperatures $0 \leq \beta_1 < \beta_2$, $\frac{Z(\beta_2)}{Z(\beta_1)}$ can be computed with a multiplicative error of ϵ with probability at least $\frac{3}{4}$ given the Hamiltonian of $\tilde{O}(\frac{n}{\epsilon^2})$ samples taken from the Gibbs distribution for inverse temperatures ranging from $[\beta_1, \beta_2]$. Furthermore, the temperature each sample is taken at can be precomputed and only depends on n , h , and $|\mathcal{A}|$.*

The theorem implies that we can approximate $Z(\beta)$, as long as $Z(\beta')$ is known in advance for some $\beta' \geq 0$ and that we can draw $\tilde{O}(\frac{n}{\epsilon^2})$ samples for inverse temperatures in the range $[\min(\beta, \beta'), \max(\beta, \beta')]$.

2.3 Pairwise Markov Random Fields

The traditional way to sample from a Gibbs distribution is using a rapidly mixing Markov chain. We begin from an arbitrary state (a graph labeling) and simulate transitions until the chain is *mixed*, meaning that the current state is roughly drawn from the desired distribution. Here we will focus on sampling from the probability space defined by a pairwise Markov random field. A pairwise Markov random field on a graph consists of vertex constraints and edge constraints.

¹ We treat the infinite case in the sense of the limit as $\beta \rightarrow \infty$.

For each vertex v , we have a constraint $b_v : \mathcal{A} \rightarrow \mathbb{R}_{\geq 0}$. For each edge $e = (u, v)$ we also have a constraint $A_e : \mathcal{A} \times \mathcal{A} \rightarrow \mathbb{R}_{\geq 0}$. Here the edges are undirected, but the function A_e is not necessarily symmetric. The probability of graph vertex labeling l is defined to be proportional to

$$\prod_v b_v(l(v)) \cdot \prod_{e=(u,v)} A_e(l(u), l(v))$$

► **Definition 7** (Local Gibbs Distribution). *We call a Gibbs distribution local if*

1. *The distribution can be represented as a pairwise Markov random field.*
2. *The Hamiltonian H can be represented as a sum over sub-functions defined on each edge and vertex. $H(\sigma) = \sum_v H_v(\sigma) + \sum_{e=(u,v)} H_e(u, v)$. Furthermore, each of these subfunctions also has a codomain of $[h]$.*
3. *An element $\sigma \in \mathcal{S}$ can be constructed in the CONGESTEDCLIQUE in $O(\log n)$ rounds.*

The third condition is needed to initialize the Markov chain.

► **Example 8** (Potts Model). We can see that the Potts model is a local Gibbs distribution. For convenience we make the change of variable, $\beta = -\ln \lambda$. We choose $b_v(x) = 1$ and

$$A_e(x, y) = \begin{cases} \lambda, & \text{if } x = y \\ 1, & \text{else} \end{cases}$$

2.4 Distributed Sampling

Feng, Sun, and Yin [5] found a preliminary distributed Markov chain to sample from pairwise Markov random fields. In the case of q -colorings, this was improved simultaneously by Fischer and Ghaffari [6] and Feng, Hayes, and Yin [4]. The improved chain can be generalized beyond colorings, see [11] by Pemmaraju and Sobel for the chain explicitly given in full generality. One transition of the chain, for some yet to be determined probability p , is defined in the following pseudocode. Every vertex v has its current label X_v . For ease of notation, in the edge acceptance step, any fraction that is not well defined (because certain vertices may not be active) equals 1. Note that on some pairwise Markov random fields this chain may have a very large mixing time, $\gg \log n$ and possibly infinite.

■ **Algorithm 1** Distributed Metropolis-Hastings.

```

for each vertex  $v$  do
   $v$  becomes active with probability  $p$ 
  if  $v$  is active then
     $v$  samples a proposed label,  $\sigma_v$ , from a distribution proportional to  $b_v(\sigma)$ 
  end if
end for
for each edge  $e = (u, v)$  do
  edge  $e$  is accepted with probability  $\frac{A_e(X_u, \sigma_v)}{\max A_e} \cdot \frac{A_e(\sigma_u, X_v)}{\max A_e} \cdot \frac{A_e(\sigma_u, \sigma_v)}{\max A_e}$ 
end for
for each active vertex  $v$  do
   $X_v \leftarrow \sigma_v$ , if every edge incident to  $v$  is accepted
end for

```

3 Our Results

3.1 Fast Mixing for the Potts Model

For the Potts model, the distributed Metropolis-Hastings Markov chain has a mixing time of $O(\log \frac{n}{\delta})$. As the proof technique is standard, we defer the proof to the appendix of the full version of our paper.

3.2 Fast Simulation of Distributed Metropolis-Hastings

We would now like to estimate the partition function of a local Gibbs distribution. As described earlier, we need to know the Hamiltonian of $k = \tilde{O}(\frac{n}{\epsilon^2})$ samples from the Gibbs distribution. This can be found by running k instances of the distributed Metropolis-Hastings Markov chain. We can simulate one transition of n instances of the chain in parallel.

► **Theorem 1.** *One transition can be simulated for n instances of the distributed Metropolis-Hastings chain in $O(n^{1/3})$ rounds.*

3.2.1 Algorithm

Initialization

Each chain starts at an initial labeling. At time step t , each vertex will hold $X_{v,i,t}$, the label of vertex v in chain i . It will also sample its new proposed label, $\sigma_{v,i,t}$. In the case the vertex is inactive in a chain, $\sigma_{v,i,t} = \sqcup$.

To decide if $X_{v,i,t+1} \leftarrow \sigma_{v,i,t}$, assuming $\sigma_{v,i,t} \neq \sqcup$, v needs to know if any of its incident edges are rejected in chain i . We will use the technique from Dolev, Lenzen, and Peled [3] for triangle counting and also used by Censor-Hillel, Kaski, Korhonen, Lenzen, Paz, Suomela [2] for matrix multiplication over semirings.

For convenience, we will use 0 based indexing. Let A be the adjacency matrix of the graph. Consider an $n \times n \times n$ cube, Q , where $Q_{v,w,i}$ is true iff $A_{v,w} = 1$ and $\{v, w\}$ is not accepted in chain i . Furthermore, think of Q evenly divided into n , $n^{2/3} \times n^{2/3} \times n^{2/3}$ subcubes. Let subcube $Q[x, y, z]$ refer to the subcube over the indices $[xn^{2/3}, (x+1)n^{2/3}] \times [yn^{2/3}, (y+1)n^{2/3}] \times [zn^{2/3}, (z+1)n^{2/3}]$. We will use a similar notation for matrices, $A[x, y]$ refers to the submatrix over the indices $[xn^{2/3}, (x+1)n^{2/3}] \times [yn^{2/3}, (y+1)n^{2/3}]$.

Computing Q

Each subcube is arbitrarily assigned to a machine that will be responsible for generating the entries of the subcube. Because edges are non-directional, we assign the subcubes $Q[x, y, z]$ and $Q[y, x, z]$ to the same machine. Each machine is assigned at most two subcubes.

Consider the machine M holding the subcubes $Q[x, y, z]$ and $Q[y, x, z]$. To fill in the subcubes, M needs to hold $A[x, y]$, $\sigma[x, z]$, $\sigma[y, z]$, $X[x, z]$, and $X[y, z]$. Altogether, M needs to receive $\Theta(n^{4/3})$ words. For every machine to do this step in parallel, each machine needs to send and receive $\Theta(n^{4/3})$ words. This communication is done in $\Theta(n^{1/3})$ rounds, using Lenzen's result [9]. M then uses the information it received in addition to its randomness to fill in the (possibly two) subcube(s) it is responsible for. More precisely, M runs the edge acceptance step of the Markov chain for the edge-chain pairs it is responsible for and fills in the subcube(s).

Updating Labels

M will compress each subcube it holds into a submatrix by applying union along the y -axis. In particular, the subcube $Q[x, y, z]$ is flattened into the submatrix $R^y[x, z]$, where $R^y[x, z]_{i,k} = \text{true}$ if there exists j such that $Q[x, y, z]_{i,j,k} = \text{true}$.

Finally, vertex u accepts its proposal in chain i if $\bigvee_y R^y_{u,i} = \text{false}$, as this indicates that no incident edge to u was rejected. To compute this for every chain, u receives $R^y_{u,i}$ for all y and i . Doing this simultaneously for all vertices, requires each machine to send and receive $O(n^{4/3})$ words of information. This takes a final $O(n^{1/3})$ rounds.

► **Corollary 9.** *With success probability at least $\frac{3}{4}$, we can approximate the partition function, $Z(\beta)$, of a local Gibbs distribution with multiplicative error ϵ in $\tilde{O}(\frac{n^{1/3}}{\epsilon^2})$ rounds, when $Z(\beta')$ is known for some β' and the associated distributed Metropolis-Hastings chain has an $O(\log n)$ mixing time for inverse temperatures in the interval $[\min(\beta, \beta'), \max(\beta, \beta')]$.*

Proof. Theorem 6 and Theorem 1 almost immediately give us the result. We draw $\tilde{O}(\frac{n}{\epsilon^2})$ samples from the Gibbs distribution by simulating $O(\log n)$ transitions of the Markov chain. There is one slight problem, the samples are held in a distributed fashion and we need the Hamiltonian of each sample held centrally at some machine. We use a very similar strategy as before, with a slight change described in our full paper. ◀

► **Corollary 2.** *There is an algorithm for approximating the number of q -colorings of a graph in $\tilde{O}(\frac{n^{1/3}}{\epsilon^2})$ rounds, when $q > \alpha\Delta$, for $\alpha > 2$, within multiplicative error ϵ , with probability at least $\frac{3}{4}$.*

3.3 Lower Bound

We can show that even in the case where the distributed Metropolis-Hastings chain has mixing time $O(\log \frac{n}{\delta})$, we still have conditional hardness results.

Consider the following Gibbs distribution. $\mathcal{A} = V \cup [3n]$. $\mathcal{S}$ are the labelings such that every vertex's label is either in $[3n]$ or is a neighboring vertex. We let the Hamiltonian count the number of edges $\{u, v\}$ such that the labels of u and v are the same vertex. It is not hard to show that this is a local Gibbs distribution and that the mixing time of the distributed Metropolis-Hastings chain is $O(\log \frac{n}{\delta})$ for $\beta \geq 0$.

As always, when $\beta = 0$, all labelings in $\mathcal{S}$ are equally likely. $Z(0) = \prod_v 3n + \deg(v)$. On the other hand, when $\beta = \infty$, we remove labelings where two neighboring vertices are labeled the same mutually neighboring vertex. This can only happen if there is a triangle in the graph. Thus, $Z(0) > Z(\infty)$ if and only if there is a triangle in the graph. In fact, if the graph has at least one triangle, a, b, c , then $1 - \frac{Z(\infty)}{Z(0)} \geq \frac{1}{16n^2}$, since that is a lower bound on the probability that vertices a and b both choose c in a uniform random labeling. Rearranging, $Z(\infty) \leq Z(0)(1 - \frac{1}{16n^2}) \leq Z(0)(1 - \frac{1}{32n^2})^2$.

Note that we can easily compute $Z(0)$ exactly. We can choose $\epsilon = \frac{1}{32n^2}$ and take an approximate count of $Z(\infty)$, $\bar{Z}$. Suppose the graph doesn't have a triangle. Then, $Z(\infty) = Z(0)$. Thus, $\bar{Z} \geq (1 - \frac{1}{32n^2})Z(0)$. Otherwise, the graph has a triangle. Then, $\bar{Z} \leq (1 + \frac{1}{32n^2})Z(\infty) \leq (1 - \frac{1}{32n^2})^2(1 + \frac{1}{32n^2})Z(0) < (1 - \frac{1}{32n^2})Z(0)$. Thus, we can use $\bar{Z}$ to determine if the graph has a triangle.

► **Theorem 10.** *Approximating the partition function of a local Gibbs distribution when $\epsilon \leq \frac{1}{32n^2}$ with success probability p is as hard as triangle detection with success probability p . This is true even when samples can be taken for all $\beta \geq 0$ in $\tilde{O}(1)$ rounds.*

Note that the best known algorithm for triangle detection takes $O(n^{0.157})$ rounds [2, 14].

References

- 1 Ivona Bezáková, Daniel Štefankovič, Vijay V. Vazirani, and Eric Vigoda. Accelerating simulated annealing for the permanent and combinatorial counting problems. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, SODA '06, pages 900–907, USA, 2006. Society for Industrial and Applied Mathematics. URL: <http://dl.acm.org/citation.cfm?id=1109557.1109656>.
- 2 Keren Censor-Hillel, Petteri Kaski, Janne H. Korhonen, Christoph Lenzen, Ami Paz, and Jukka Suomela. Algebraic methods in the congested clique. In *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing*, PODC '15, pages 143–152, New York, NY, USA, 2015. Association for Computing Machinery. doi:10.1145/2767386.2767414.
- 3 Danny Dolev, Christoph Lenzen, and Shir Peled. "tri, tri again": finding triangles and small subgraphs in a distributed setting. In *Proceedings of the 26th International Conference on Distributed Computing*, DISC'12, pages 195–209, Berlin, Heidelberg, 2012. Springer-Verlag. doi:10.1007/978-3-642-33651-5_14.
- 4 Weiming Feng, Thomas P. Hayes, and Yitong Yin. Distributed symmetry breaking in sampling (optimal distributed randomly coloring with fewer colors), 2018. [arXiv:1802.06953](https://arxiv.org/abs/1802.06953).
- 5 Weiming Feng, Yuxin Sun, and Yitong Yin. What can be sampled locally? In *Proceedings of the ACM Symposium on Principles of Distributed Computing*, PODC '17, pages 121–130, New York, NY, USA, 2017. Association for Computing Machinery. doi:10.1145/3087801.3087815.
- 6 Manuela Fischer and Mohsen Ghaffari. A Simple Parallel and Distributed Sampling Technique: Local Glauber Dynamics. In Ulrich Schmid and Josef Widder, editors, *32nd International Symposium on Distributed Computing (DISC 2018)*, volume 121 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 26:1–26:11, Dagstuhl, Germany, 2018. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.DISC.2018.26.
- 7 Mark Jerrum. A very simple algorithm for estimating the number of k-colorings of a low-degree graph. *Random Struct. Algorithms*, 7(2):157–165, September 1995. doi:10.1002/RS.3240070205.
- 8 Mark R. Jerrum, Leslie G. Valiant, and Vijay V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science*, 43:169–188, 1986. doi:10.1016/0304-3975(86)90174-X.
- 9 Christoph Lenzen. Optimal deterministic routing and sorting on the congested clique. In *Proceedings of the 2013 ACM Symposium on Principles of Distributed Computing*, PODC '13, pages 42–50, New York, NY, USA, 2013. Association for Computing Machinery. doi:10.1145/2484239.2501983.
- 10 Hongyang Liu, Yitong Yin, and Yiyao Zhang. Work-efficient parallel counting via sampling, 2024. doi:10.48550/arXiv.2408.09719.
- 11 Sriram V. Pemmaraju and Joshua Z. Sobel. Exact distributed sampling. In *Structural Information and Communication Complexity: 30th International Colloquium, SIROCCO 2023, Alcalá de Henares, Spain, June 6–9, 2023, Proceedings*, pages 558–575, Berlin, Heidelberg, 2023. Springer-Verlag. doi:10.1007/978-3-031-32733-9_25.
- 12 L.G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189–201, 1979. doi:10.1016/0304-3975(79)90044-6.
- 13 Daniel Štefankovič, Santosh Vempala, and Eric Vigoda. Adaptive simulated annealing: A near-optimal connection between sampling and counting. *J. ACM*, 56(3), May 2009. doi:10.1145/1516512.1516520.
- 14 Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu, and Renfei Zhou. New bounds for matrix multiplication: from alpha to omega. In *ACM-SIAM Symposium on Discrete Algorithms*, 2023. URL: <https://api.semanticscholar.org/CorpusID:259937341>.