Optimal and Efficient Partite Decompositions of Hypergraphs

Andrew Krapivin
Carnegie Mellon University
andrew@krapivin.net

Benjamin Przybocki
Carnegie Mellon University
benjamin.przybocki@gmail.com

Nicolás Sanhueza-Matamala
Universidad de Concepción
nsanhuezam@udec.cl

Bernardo Subercaseaux
Carnegie Mellon University
bersub@cmu.edu

Abstract

We study the problem of partitioning the edges of a d-uniform hypergraph H into a family $\mathcal{F}$ of complete d-partite hypergraphs (d-cliques). We show that there is a partition $\mathcal{F}$ in which every vertex $v \in V(H)$ belongs to at most $(\frac{1}{d!} + o_d(1))n^{d-1}/\lg n$ members of $\mathcal{F}$. Together with a simple information-theoretic lower bound, this settles the central question of a line of research initiated by Erdős and Pyber (1997) for graphs, and more recently by Csirmaz, Ligeti, and Tardos (2014) for hypergraphs. The $d = 2$ case of this theorem answers a 40-year-old question of Chung, Erdős, and Spencer (1983). Furthermore, our construction is algorithmically efficient: such optimal partitions can be constructed in time $O(n^d/d!)$. An immediate corollary of our result is an improved upper bound for the maximum share size for binary secret sharing schemes on uniform hypergraphs.

Building on results of Nechiporuk (1969), we prove that every graph with fixed edge density $\gamma \in (0, 1)$ has a biclique partition of total weight at most $(\frac{1}{2} + o(1)) \cdot h_2(\gamma)\frac{n^2}{\lg n}$, where h_2 is the binary entropy function. This result is asymptotically tight and answers a further question of Chung, Erdős, and Spencer. Our construction implies that such biclique partitions can be constructed in time $O(m)$, which answers a question of Feder and Motwani (1995) and also improves upon results of Mubayi and Turán (2010) as well as Chavan, Rabinia, Grosu, and Brocanelli (2025). Using similar techniques, we also give an $n^{1+o(1)}$ algorithm for finding a subgraph $K_{t,t}$ with $t = (1 - o(1))\frac{\gamma}{h_2(\gamma)} \lg n$, which matches the celebrated Kővári–Sós–Turán guarantee for small γ.

Our results show that biclique partitions are information-theoretically optimal representations for graphs at every fixed density, which makes them a natural succinct data structure. We show that with this succinct representation one can answer independent set queries and cut queries in time $O(n^2/\lg n)$; prior work of Bansal, Williams, and Vassilevska Williams gave subquadratic algorithms for independent set queries at the cost of $\omega(n^2)$ preprocessing. We also show that if we increase the space usage by a constant factor, we can compute a modification of Charikar's 2-approximation algorithm for the densest subgraph problem that runs in time $O(n^2/\lg \alpha)$ and gives a 2α-approximation for any $\alpha > 1$, thus establishing the first approximation guarantees that can be obtained in subquadratic time.

Finally, we show that graphs with *polynomially bounded shattering*, a class including graphs of bounded VC-dimension, admit biclique partitions of weight $O(n^{2-1/(d+1)})$, where d is the shattering exponent, which extends recent results of Cardinal and Yuditsky (2025).

Contents

1 Introduction

In 1956, Lupanov [Lup56] considered the problem of minimizing the number of wires in a directed circuit, and observed that if a input ports were fully connected to b output ports, then the $a \cdot b$ wires could be replaced by adding one intermediary *switch s* to reduce the number of wires down to $a + b$. A repeated application of this idea is illustrated in Figure 1.
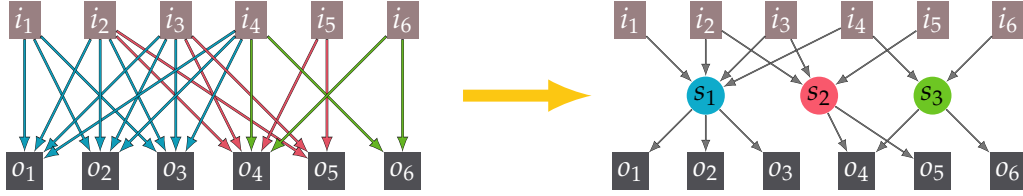

Figure 1: Illustration of the application of biclique partitions in circuit design.

Without switches, a circuit with n input ports and n output ports may require n^2 wires, but Lupanov proved that with switches, such a circuit can be realized with only $(1 + o(1))\frac{n^2}{\lg n}$ wires.[1] While Lupanov formulated his result in terms of circuits, it can be naturally recast in terms of biclique partitions of bipartite graphs.

Theorem 1 (Lupanov [Lup56]). *Let $G = (X \sqcup Y, E)$ be a bipartite graph with $|X| \geqslant |Y|$ and $|Y| = \omega(\lg |X|)$. Then, G can be partitioned into bicliques $B_1, \ldots B_k$ with weight $\sum_{i=1}^{k} |V(B_i)| \leqslant (1 + o(1))\frac{|X| \cdot |Y|}{\lg |X|}$. Moreover, there are some graphs G for which every biclique partition has weight $(1 - o(1))\frac{|X| \cdot |Y|}{\lg |X|}$.*

The lower bound of Theorem 1, applied to a bipartite graph with $|X| = |Y| = n/2$, yields a general lower bound of $(1 - o(1))\frac{n/2 \cdot n/2}{\lg n} = \left(\frac{1}{4} - o(1)\right)\frac{n^2}{\lg n}$ for n-vertex graphs. On the other hand, Lupanov's upper bound can be easily extended to general graphs by the following observation: from a graph $G = (V, E)$, choose an arbitrary total order $\prec$ on V, and then create a bipartite graph $G' = (V_1 \sqcup V_2, E')$ where each part V_i is a copy of V, and for each edge $\{u, v\} \in E$ with $u \prec v$, add to G' the edge $\{u_1, v_2\}$ where $u_1 \in V_1$ is the element corresponding to u, and $v_2 \in V_2$ the element corresponding to v. Then, observe that bicliques in G are in a one-to-one correspondence to bicliques in G', and thus Theorem 1 applied to G' implies that G admits a biclique partition of weight $(1 + o(1))\frac{n^2}{\lg n}$, with $n := |V(G)|$. Therefore, if we denote by $\mathsf{BP}(G)$ the minimum weight of a biclique partition of a graph G, and then $\mathsf{BP}(n) := \max\{\mathsf{BP}(G) : G \text{ a graph with } |V(G)| = n\}$, Lupanov's result has the following consequence.

Corollary 2. $\left(\frac{1}{4} - o(1)\right)\frac{n^2}{\lg n} \leqslant \mathsf{BP}(n) \leqslant (1 + o(1))\frac{n^2}{\lg n}$.

Unfortunately, Lupanov's result, originally published in Russian, continues to remain largely unknown in the West [Juk12], and it was only in 1983 that Chung, Erdős, and Spencer published bounds for general graphs, unaware of Lupanov's previous work. Denoting by $\mathsf{BC}(G)$, and analogously, $\mathsf{BC}(n)$, the minimum weight of a biclique *covering* (i.e., edges of G may belong to multiple bicliques), we trivially have $\mathsf{BC}(n) \leqslant \mathsf{BP}(n)$, and the Chung–Erdős–Spencer (CES) theorem states:

Theorem 3 ([CES83]). $\left(\frac{\lg e}{2e} - o(1)\right)\frac{n^2}{\lg n} \leqslant \mathsf{BC}(n) \leqslant \mathsf{BP}(n) \leqslant \left(\frac{\lg e}{2} + o(1)\right)\frac{n^2}{\lg n}$.

---

[1]All logarithms in this paper are base 2.

1

Since, $\lg(e)/(2e) \approx 0.265 > 1/4$, and $\lg(e)/2 \approx 0.721 < 1$, Theorem 3 improved Corollary 2 on both accounts. Tuza [Tuz84] and Bublitz [Bub86] independently proved similar results, albeit with worse quantitative bounds. Chung, Erdős, Spencer, and Tuza asked whether $\lim_{n\to\infty} \mathsf{BP}(n)/(n^2/\lg(n))$ and $\lim_{n\to\infty} \mathsf{BC}(n)/(n^2/\lg(n))$ exist and if so what their values are. Our first contribution, presented in Section 2, is an answer to these questions:

Theorem 4. $\left(\frac{1}{2} - o(1)\right)\frac{n^2}{\lg n} \leqslant \mathsf{BC}(n) \leqslant \mathsf{BP}(n) \leqslant \left(\frac{1}{2} + o(1)\right)\frac{n^2}{\lg n}$.

In fact, in Section 8, we show that the same bounds hold for $\mathsf{BC}(G)$ and $\mathsf{BP}(G)$ for $G \sim G(n, 1/2)$ with high probability, which also answers a question of Chung, Erdős, and Spencer.

An important extension of the CES theorem is the Erdős–Pyber theorem [EP97], which says that every graph can be partitioned into bicliques so that each vertex is in at most $O(n/\lg n)$ bicliques, a result that directly implies that $\mathsf{BP}(n) = O(n^2/\lg n)$. For example, in the aforementioned circuit minimization problem, the Erdős–Pyber theorem says that one can not only reduce the number of wires to $O(n^2/\lg n)$ but also ensure that each input/output port is connected to $O(n/\lg n)$ wires, and thus no part of the circuit is very cluttered.

Interestingly, a key application of the Erdős–Pyber theorem arises in cryptography, and more specifically, in the design of *secret sharing schemes* [BFM14, BDSGV96]. In a secret sharing scheme, the goal is to split a secret string s into pieces, and distribute them to agents, in such a way that only certain specific groups of agents have enough information to reconstruct the secret if they join their pieces. These 'allowed' groups are often all of the same size; if that size is 2, then they can be specified by edges of a graph where agents are the vertices, but for a larger size d, they are specified by d-uniform hypergraphs. In this context, covering results like the CES theorem allow us to bound the *total share* of the secret pieces used, and Erdős–Pyber theorem bounds the *maximum share* some agent is required to receive. This has motivated a line of work on Erdős–Pyber results for d-uniform hypergraphs [CLT15, Bei23].

More precisely, define the *local biclique partition number* of a graph G, denoted $\mathsf{LBP}(G)$, as the least natural number such that there is a biclique partition of G for which every vertex is in at most $\mathsf{LBP}(G)$ bicliques. Similarly, let $\mathsf{LBP}(n) := \max\{\mathsf{LBP}(G) : G \text{ a graph with } |V(G)| = n\}$, and observe that $\mathsf{BP}(n) \leqslant n \cdot \mathsf{LBP}(n)$. With this notation, the Erdős–Pyber theorem says that $\mathsf{LBP}(n) = O(n/\lg(n))$. This was improved by Csirmaz, Ligeti, and Tardos [CLT15] to $\mathsf{LBP}(n) \leqslant (1 + o(1))\frac{n}{\lg n}$. Our proof of Theorem 4 in fact shows that $\mathsf{LBP}(n) \leqslant \left(\frac{1}{2} + o(1)\right)\frac{n}{\lg n}$. Theorem 4 also implies that this is tight, since $\mathsf{LBP}(n) \geqslant \mathsf{BP}(n)/n \geqslant \left(\frac{1}{2} - o(1)\right)\frac{n}{\lg n}$.

Csirmaz, Ligeti, and Tardos also generalized the Erdős–Pyber theorem to d-uniform hypergraphs (d-graphs). Let a *d-clique* be a d-graph $H = (V_1 \sqcup \cdots \sqcup V_d, E)$ such that

$$E(H) = \{(v_1, \ldots, v_d) : v_i \in V_i \text{ for all } i \in [d]\}.$$

This is the natural generalization of a biclique to d-uniform hypergraphs. We also will use the notation $(V_1, \ldots, V_d)$ to refer to this d-clique, and say that $V_1, \ldots, V_d$ are its *parts*. Given a d-uniform hypergraph H, let $\mathsf{LMP}_d(H)$ be the least natural number such that there is a d-clique partition of H for which every vertex is in at most $\mathsf{LMP}_d(H)$ d-cliques. We define $\mathsf{MP}_d(H)$, $\mathsf{MC}_d(H)$, $\mathsf{MP}_d(n)$, $\mathsf{MC}_d(n)$, and $\mathsf{LMP}_d(n)$ analogously to the definitions of BC and BP above. Csirmaz, Ligeti, and Tardos proved that

$$\left(\frac{0.53}{d!} - o_d(1)\right)\frac{n^{d-1}}{\lg n} \leqslant \mathsf{LMP}_d(n) \leqslant \left(\frac{1}{(d-2)!} + o_d(1)\right)\frac{n^{d-1}}{\lg n}.$$

Earlier, Bublitz [Bub86] proved that $\mathsf{MP}_d(n) \leqslant \left(\frac{32}{(d-1)!} + o_d(1)\right)\frac{n^d}{\lg n}$. Our main theorem, proved in Section 3, improves on these results and completely settles the asymptotics of the relevant functions:

2

Theorem 5. $\left(\frac{1}{d!} - o_d(1)\right)\frac{n^{d-1}}{\lg n} \leqslant \frac{MC_d(n)}{n} \leqslant \frac{MP_d(n)}{n} \leqslant LMP_d(n) \leqslant \left(\frac{1}{d!} + o_d(1)\right)\frac{n^{d-1}}{\lg n}$.

Moreover, we show that a d-clique partition realizing the upper bound on $\mathsf{LMP}_d(n)$ can be constructed deterministically in time $O(n^d/d!)$. Using Stinson's decomposition theorem [Sti94] as discussed in [CLT15], an immediate consequence of our upper bound on $\mathsf{LMP}_d(n)$ is an improved upper bound for the maximum share size for secret sharing schemes with binary secrets on uniform hypergraphs.

Section 4 is devoted to biclique partitions that consider the edge density $\gamma := |E(G)|/\binom{|V(G)|}{2}$ of the input graph; both for dense graphs $\gamma > 1/2$, and sparse graphs $\gamma < 1/2$ it is possible to obtain smaller biclique partitions than those implied by Theorem 4. In their 1983 paper, Chung, Erdős, and Spencer also asked about the quantities $\mathsf{BC}(n,\gamma) := \max\{\mathsf{BC}(G) : G \text{ a graph with } |V(G)| = n \text{ and } |E(G)| = \gamma\binom{n}{2}\}$ and the similarly defined $\mathsf{BP}(n,\gamma)$. Unbeknownst to Chung, Erdős, and Spencer, Nechiporuk [Nec69] solved the analogous problems for bipartite graphs using a clever generalization of Lupanov's construction. Let $h_2(x) := -x\lg(x) - (1-x)\lg(1-x)$ be the binary entropy function.

Theorem 6 (Nechiporuk [Nec69])**.** *Let $G = (X \sqcup Y, E)$ be a bipartite graph with $|Y| \geqslant |X|$, and let $\gamma \in (0,1)$ be such that $|X| = \omega\left(\frac{\lg|Y|}{h_2(\gamma)}\right)$, $\max\{\gamma^{-1}, (1-\gamma)^{-1}\} = |Y|^{o(1)}$, and $|E| = \gamma|X||Y|$. Then, G admits a biclique partition of weight $(1 + o(1)) \cdot h_2(\gamma)\frac{|X||Y|}{\lg|Y|}$, and this is asymptotically best possible.*

By the same reasoning as above, this implies that $\left(\frac{1}{4} - o(1)\right) \cdot h_2(\gamma)\frac{n^2}{\lg n} \leqslant \mathsf{BP}(n,\gamma) \leqslant (1 + o(1)) \cdot h_2(\gamma)\frac{n^2}{\lg n}$ for $\max\{\gamma^{-1}, (1-\gamma)^{-1}\} = n^{o(1)}$. Building on Nechiporuk's construction, we solve the problem asymptotically in Section 4:

Theorem 7. *Let $\gamma \in (0,1)$ be such that $\max\{\gamma^{-1}, (1-\gamma)^{-1}\} = n^{o(1)}$. Then, $BP(n,\gamma) \sim \frac{h_2(\gamma)}{2} \cdot \frac{n^2}{\lg n}$. Furthermore, given a graph represented as an adjacency list, a biclique partition realizing the upper bound can be constructed deterministically in time $O(m)$.*

In Section 4.1, we also prove a variant of this result for graphs with bounds on the degrees of each vertex, which can be seen as blending Theorem 7 with the Erdős–Pyber theorem. Specifically, we show that if G is a regular graph with edge density γ, then $\mathsf{LBP}(G) \leqslant \left(\frac{1}{2} + o(1)\right) \cdot h_2(\gamma)\frac{n}{\lg n}$; the actual result we prove, Theorem 26, is a bit stronger than this.

A decomposition of a graph into bicliques can be fruitfully viewed as a compressed representation of the graph, and the above theorems provide theoretical guarantees on the quality of such a compression scheme. Feder and Motwani's seminal work showed that the runtime of several graph algorithms—including those for matchings, vertex- and edge-connectivity, and shortest paths—can be asymptotically improved by operating on the compressed representation of a graph [FM95]. The effectiveness of this approach depends on two parameters: *runtime* (i.e., the complexity of computing a biclique partition) and *quality* (i.e., the weight of the resulting partition). Feder and Motwani developed a deterministic polynomial-time algorithm for constructing biclique partitions of weight $O(h_2(\gamma)n^2/\lg n)$, and similarly, Mubayi and Turán [MT10] independently gave a deterministic polynomial-time algorithm for constructing biclique partitions of weight $O(n^2/\lg n)$. The algorithm of Feder and Motwani was recently improved by Chavan, Rabinia, Grosu, and Brocanelli [CRGB25]. In all three cases, the runtime is $\omega(n^2)$ and depends on the desired weight of the biclique partition. While each algorithm produces a biclique partition of weight $O(n^2/\lg(n))$, none produces a biclique decomposition with the strong quantitative bounds given by the CES theorem or Nechiporuk's theorem. It seems to have been hitherto unnoticed that Lupanov's proof

3

of Theorem 1 yields a deterministic $O(n^2)$ algorithm for constructing a biclique partition of weight $O(n^2/\lg(n))$, one that is much simpler than the previous algorithms and achieves the upper bound stated in Corollary 2; similarly, Nechiporuk's proof of Theorem 6 yields a deterministic $O(m)$ algorithm for constructing a biclique partition of weight $O(h_2(\gamma)n^2/\lg n)$, which answers a question from Feder and Motwani's paper. Our proofs of Theorems 4 and 7 similarly yield deterministic $O(n^2)$ and $O(m)$ algorithms respectively producing a biclique partition with the optimal bound, improving on all of the previous results and achieving optimality both with respect to runtime and quality.

Building on the idea of using biclique partitions as a form of graph compression, in Section 5 we further investigate on their usage as succinct or compact data structures; a data structure for a class of objects U is said to be *succinct* if it uses $(1 + o(1))\lg(U)$ bits of space and *compact* if it uses $O(\lg(U))$ bits. Since the number of graphs on n vertices with density γ is $\binom{\binom{n}{2}}{\gamma\binom{n}{2}}$ and $\lg\left(\binom{\binom{n}{2}}{\gamma\binom{n}{2}}\right) \sim \frac{h_2(\gamma)n^2}{2}$, it follows that a representation for graphs is succinct if it uses $\frac{h_2(\gamma)n^2}{2}$ bits, which is precisely what Theorem 7 allows us to do given that a vertex can be represented using $\lg n$ bits. This result gives theoretical support to more applied work in graph compression [FGK+22, HN13]. Furthermore, we show that if we store graphs in succinct space as biclique partitions, certain natural graph queries can be answered faster than using a (compressed) adjacency matrix. Namely, we show that given a subset S of the vertices, we can decide if it is an independent set in time $O(n^2/\lg n)$, as opposed to the naive $\Theta(n^2)$ algorithm. While better asymptotic algorithms exist for this task [Wil07, BW09, WW18], they require $\omega(n^2)$ preprocessing (in fact, the two latter algorithms use Feder and Motwani's algorithm as a subroutine) and use $\omega(n^2)$ space. We also show that the succinct biclique representation allows to answer *cut queries* in time $O(n^2/\lg n)$. In a cut query the input consists of two disjoint subsets of vertices S and T, and the output shall be the number of edges between S and T. The main result of Section 5 is that, by maintaining 'back references' from each vertex to the list of bicliques it belongs to, which leads to a compact data structure (*CB representation*), we can approximate the densest subgraph problem (i.e., find nonempty $S \subseteq V$ that maximizes $|E(G[S])|/|S|$) in subquadratic time:

Theorem 8. *Given a CB representation for an n-vertex graph G, where the biclique partition has weight $O(n^2/\lg n)$, and any $\alpha > 1$, one can compute a 2α-approximation for the densest subgraph problem in time $O(n^2/\lg \alpha)$.*

While the densest subgraph problem is polynomial-time solvable [Cha00], in practice very fast approximation algorithms are used [LMFB24]; ours is, to the best of our knowledge, the first one with a subquadratic runtime (e.g., Theorem 8 yields a $2\lg\lg n$-approximation in time $O(n^2/\lg\lg\lg n) = o(n^2)$).

Section 6 is dedicated to algorithms for finding large balanced bicliques. Cardinal and Yuditsky [CY25] observed that, as a corollary of the existence of biclique partitions of weight $O(n^2/\lg n)$, every graph with $\Omega(n^2)$ edges contains a biclique $K_{t,t}$ with $t = \Omega(\lg n)$ (see Lemma 36). This is an instance of the celebrated Kővári–Sós–Turán (KST) theorem [KST54] (see Theorem 34). In fact, the proof of the CES theorem (Thm. 3), and the similar proofs given by Tuza [Tuz84] and Bublitz [Bub86], are based on repeated applications of the KST theorem to peel off large bicliques from a graph until the edge density is sufficiently small. Efficiently finding large bicliques in a sufficiently dense graph plays a crucial role in the algorithms of Feder and Motwani [FM95], Mubayi and Turán [MT10], and Chavan, Rabinia, Grosu, and Brocanelli [CRGB25]. Feder and Motwani's algorithm implies that in a graph with $\Omega(n^2)$ edges, a biclique $K_{t,t}$ with $t = \Omega(\lg n)$ can be found deterministically in time $O(n^2 \log n)$, while Chavan, Rabinia, Grosu, and Brocanelli's

Theorem 26
Erdős–Pyber density

$$\mathsf{LBP}(n,\gamma) \sim \frac{h_2(\gamma)}{2}\frac{n}{\lg n}$$

— implies →

Theorem 11
Optimal Erdős–Pyber

$$\mathsf{LBP}(n) \sim \frac{1}{2}\frac{n}{\lg n}$$
$O(n^2)$ construction

— base case →

Theorem 18
Opt. Erdős–Pyber hypergraphs

$$\mathsf{LMP}_d(n) \sim \frac{1}{d!}\frac{n^{d-1}}{\lg n}$$
$O(n^k)$ construction

Theorem 7
Optimal CES Density

$$\mathsf{BP}(n,\gamma) \sim \frac{h_2(\gamma)}{2}\frac{n^2}{\lg n}$$
$O(m)$ construction

— implies →

Theorem 4
Optimal CES

$$\mathsf{BP}(n) \sim \mathsf{BC}(n) \sim \frac{1}{2}\frac{n^2}{\lg n}$$
$O(n^2)$ construction

— base case →

Theorem 14
Opt. CES hypergraphs

$$\mathsf{MP}_d(n) \sim \mathsf{MC}_d(n) \sim \frac{1}{d!}\frac{n^d}{\lg n}$$
$O(n^k)$ construction

Theorem 40
Dense random $\mathcal{G}(n,p)$ graphs

$$\mathbb{P}\left[\mathsf{BP}(G) \sim \frac{h_2(p)}{2}\frac{n^2}{\lg n}\right] = 1$$

Theorem 9
Fast Large Bicliques

We find $K_{t,t}$ with $t \sim \frac{\gamma \lg n}{h_2(\gamma)}$
in time $O(nt/\gamma)$

Theorem 39
Bounded shattering

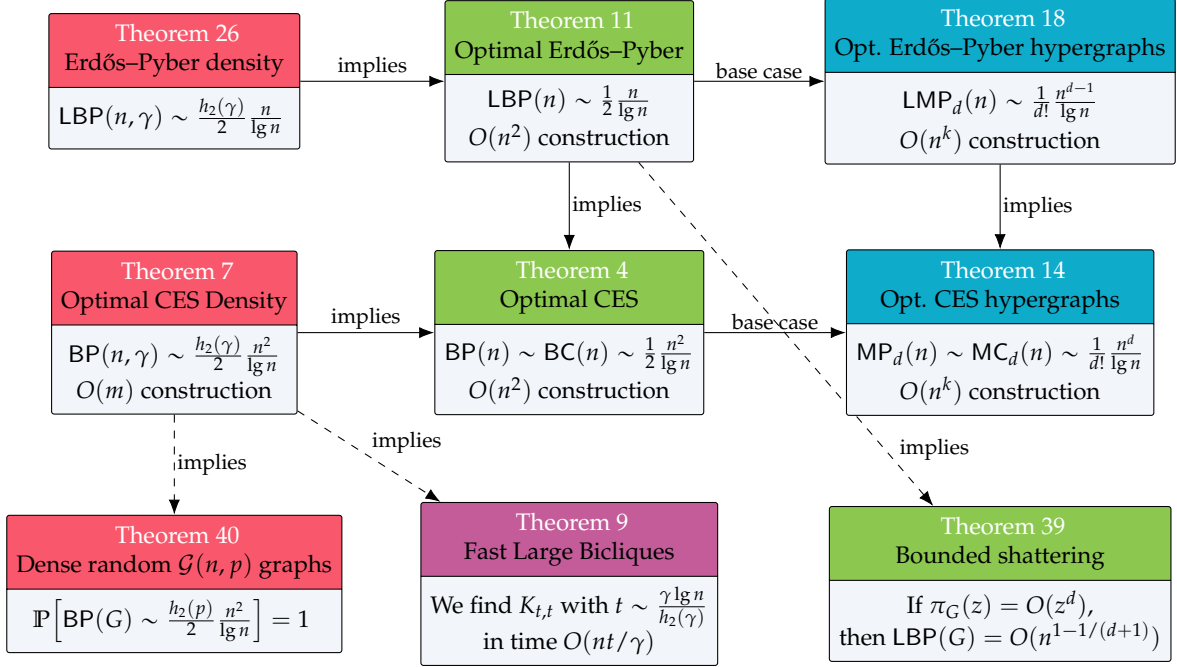If $\pi_G(z) = O(z^d)$,
then $\mathsf{LBP}(G) = O(n^{1-1/(d+1)})$

Figure 2: Summary of our results. Dashed implications represent that other tools are also required. (■ := density-aware, ■ := graph partitions, ■ := hypergraph partitions, ■ := finding bicliques)

work improves this to $O(n^2)$. Mubayi and Turán's algorithm gives a stronger quantitative guarantee: they give a deterministic $O(m)$ that finds a biclique $K_{t,t}$ with $t = \left(\frac{1}{5\lg(4\cdot e/\gamma)} - o(1)\right)\lg(n)$ in a graph with edge density γ; they also give a deterministic $O(n^{2.42})$ that finds a biclique $K_{t,t}$ with $t = \left(\frac{1}{\lg(4\cdot e/\gamma)} - o(1)\right)\lg(n)$. In fact, their first algorithm can be made to run in time $O(n^{1.42})$ in a model in which constant-time degree queries are allowed. As a corollary of our constructive proof of Theorem 7, we obtain a deterministic $O(m)$ algorithm that finds a biclique $K_{t,t}$ with $t = (1 - o(1))\frac{\gamma \lg(n)}{h_2(\gamma)}$, which is better than Mubayi and Turán's algorithm. In fact, a more refined analysis allows us to show the following:

Theorem 9. *Given a graph with edge density γ such that $\max\{\gamma^{-1}, (1-\gamma)^{-1}\} = n^{o(1)}$, there is a randomized $O\left(\frac{n\lg n}{h_2(\gamma)}\right) = n^{1+o(1)}$ time algorithm that returns a biclique $K_{t,t}$ with $t = (1 - o(1))\frac{\gamma \lg n}{h_2(\gamma)}$ with high probability. Furthermore, if we are allowed to query the degree of a vertex in $O(1)$ time, the algorithm can be made deterministic.*

Finally, in Section 7, we show that our construction used to prove that $\mathsf{LBP}(n) \leqslant \left(\frac{1}{2} + o(1)\right)\frac{n}{\lg n}$ can be naturally adapted to prove stronger bounds for graphs with *polynomially bounded shattering*, a notion closely related to VC-dimension. This result strengthens and generalizes bounds on $\mathsf{BC}(G)$ and $\mathsf{BP}(G)$ for d-dimensional semi-algebraic graphs with bounded complexity [Do19, AAE+25, CY25], a family of graphs that includes many intersection graphs of geometric objects. As discussed in those references, biclique covers for such graphs have numerous applications in computational geometry, such as range searching and intersection queries for sets of geometric objects in $\mathbb{R}^d$.
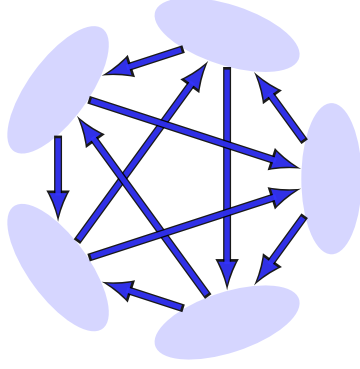
(a) Almost-regular tournament on the parts.
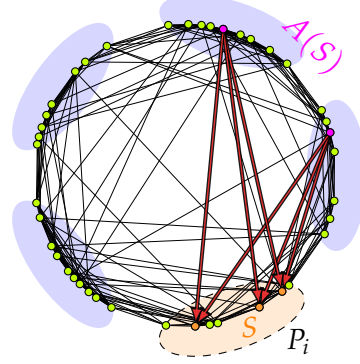
(b) Oriented biclique $(S, A(S))$ for a subset $S \subset P_i$.

Figure 3: Illustration of the biclique partition construction.

2 Optimal biclique partitions

Recall that, given a graph[2] $G = (V, E)$ a *biclique partition* is a set $\mathcal{B} := \{B_1, \ldots, B_k\}$ where each B_i is a bipartite graph, and such that $E(G) = \bigsqcup_{i=1}^{k} E(B_i)$. The *weight* $w(\mathcal{B})$ of such a partition is simply $w(\mathcal{B}) := \sum_{i=1}^{k} |V(B_i)|$. The *load* of a vertex v, denoted $\ell_{\mathcal{B}}(v)$, is the number of bicliques B_i such that $v \in V(B_i)$. Therefore, for any partition $\mathcal{B}$ of a graph G on n vertices, we have

$$w(\mathcal{B}) = \sum_{B \in \mathcal{B}} |V(B)| = \sum_{B \in \mathcal{B}} \sum_{v \in V(G)} \mathbb{1}_{[v \in V(B)]} = \sum_{v \in V(G)} \sum_{B \in \mathcal{B}} \mathbb{1}_{[v \in V(B)]} = \sum_{v \in V(G)} \ell_{\mathcal{B}}(v) \leqslant n \cdot \max_{v \in V(G)} \ell_{\mathcal{B}}(v),$$

from where $\mathsf{BP}(G) \leqslant n \cdot \mathsf{LBP}(G)$ and thus $\mathsf{BP}(n) \leqslant n \cdot \mathsf{LBP}(n)$.

2.1 Upper bound

First, we will prove the upper bound on Theorem 4. In fact, we will show an optimal upper bound on $\mathsf{LBP}(n)$, and then use the inequality $\mathsf{BP}(n) \leqslant n \cdot \mathsf{LBP}(n)$. We shall need the following concept.

Definition 10. We say that a tournament (i.e., an orientation of the complete graph) is *almost regular* if, for every vertex, the indegree and outdegree differ by at most 1.

It is not hard to see that for every n, one can efficiently construct an almost-regular tournament on n vertices. For example, given $i, j \in [n]$, let $d_n(i, j) = \min(|i - j|, n - |i - j|)$. Then, the directed graph T_n with $V(T_n) = [n]$ and

$$E(T_n) := \{(i, j) : (i < j \text{ and } d_n(i, j) \equiv 0 \pmod 2) \text{ or } (i > j \text{ and } d_n(i, j) \equiv 1 \pmod 2)\}$$

is an almost regular tournament. See Figure 3a for an illustration.

Theorem 11. *We have $LBP(n) \leqslant \frac{1}{2}\frac{n}{\lg n} + O\left(\frac{n \cdot \lg \lg n}{\lg^2(n)}\right)$, and there is a biclique partition realizing this bound in which the number of bicliques is $O(n^2 / \lg^3(n))$. Furthermore, given a graph G on n vertices, such a biclique partition can be constructed deterministically in time $O(n^2)$.*

---

[2]Unless otherwise specified, all graphs we consider are finite, simple, and undirected.

Proof. Let $r := \lfloor \lg n - 2 \lg \lg n \rfloor$ and partition $V(G)$ into $\lceil n/r \rceil$ parts $P_1, \ldots, P_{\lceil n/r \rceil}$ of size at most r each. Given a vertex $v \in V(G)$, let $g(v)$ be the index of the part P_i containing v. Then, we consider an almost-regular tournament R between the parts; so for each pair of part indices $i \neq j$ from $[[n/r]]$, we write $R(i, j)$ if there is a directed edge from i to j in this tournament. Now, for each part P_i, and every nonempty subset $S \subseteq P_i$, we define $A(S) := \{v \in V : N(v) \cap P_i = S \text{ and } R(g(v), i)\}$. See Figure 3 for an illustration. We now consider the set of bicliques

$$\mathcal{C} := \underbrace{\{(S, A(S)) : S \subseteq P_i, i \in [[n/r]]\}}_{\text{across parts}} \cup \underbrace{\{(\{u\}, \{v\}) : \{u, v\} \in E(G), g(u) = g(v)\}}_{\text{within parts}}$$

First, let us see that $\mathcal{C}$ is a biclique partition of G. Let $\{u, v\} \in E(G)$. If $g(u) = g(v)$, then $\{u, v\}$ is covered uniquely by the biclique $(\{u\}, \{v\}) \in \mathcal{C}$. Otherwise, $g(u) \neq g(v)$, and we assume without loss of generality that $R(g(u), g(v))$. Let $S = N(u) \cap P_{g(v)}$. We claim that $\{u, v\}$ is covered uniquely by the biclique $(S, A(S)) \in \mathcal{C}$. Indeed, $u \in A(S)$ and $v \in S$, so $\{u, v\}$ is covered by $(S, A(S))$. Furthermore, if $\{u, v\}$ were covered by another biclique $(S', A(S')) \in \mathcal{C}$, then we must have $u \in A(S')$ and $v \in S'$, since $R(g(x), g(y))$ for every $x \in A(S')$ and $y \in S'$. Since $v \in P_{g(v)} \cap S'$, we have $S' \subseteq P_{g(v)}$. Then, since $u \in A(S')$, we have $S' = N(u) \cap P_{g(v)} = S$. Hence, $(S', A(S')) = (S, A(S))$, so $\{u, v\}$ is covered uniquely by a biclique in $\mathcal{C}$.

Now, fix a vertex v, and let us count how many bicliques from $\mathcal{C}$ contain v. First, there are the bicliques of the form $(\{u\}, \{v\})$ for $g(u) = g(v)$, of which there are at most $|P_{g(v)}| - 1 = O(\lg(n))$. Second, there are the bicliques of the form $(S, A(S))$ with $v \in S$, of which there are at most

$$O\left(2^{|P_{g(v)}|}\right) = O\left(2^{\lg n - 2 \lg \lg n}\right) = O\left(\frac{n}{\lg^2(n)}\right).$$

Third, there are the bicliques of the form $(S, A(S))$ with $v \in A(S)$. By almost regularity of R, at most $\lceil \frac{n}{2r} \rceil$ parts P_i hold $R(g(v), i)$, and thus the number of such bicliques is at most

$$\left\lceil \frac{n}{2r} \right\rceil \leq \frac{n}{2(\lg n - 2 \lg \lg n - 1)} + 1 \leq \frac{n}{2 \lg n} + O\left(\frac{n \cdot \lg \lg n}{\lg^2(n)}\right)$$

This concludes the first part of the theorem. For the total number of bicliques we have

$$|\mathcal{C}| = O\left(\frac{n}{r} \cdot (2^r + r^2)\right) = O\left(\frac{n^2}{\lg^3(n)}\right).$$

As the pseudocode in Algorithm 1 demonstrates, the above construction can be implemented in deterministic $O(n^2)$ time. $\qquad \square$

2.2 Lower bound

The lower bound on BC is based on a simple information-theoretic argument: if every n-vertex graph can be covered with bicliques of weight at most $(r + o(1))\frac{n^2}{\lg n}$, then we could specify an n-vertex graph with $(r + o(1))n^2$ bits, which is impossible if $r < \frac{1}{2}$. This argument is given in slightly less detail in [CY25].

Theorem 12. *We have*

$$BC(n) \geq \left(\frac{1}{2} - o(1)\right)\frac{n^2}{\lg n}.$$

7

Algorithm 1 Biclique Partition

Require: A graph $G = (V, E)$ with $V = [n]$.

1: $r \leftarrow \lfloor \lg n - 2 \lg \lg n \rfloor$
2: **function** $R(i, j) \triangleright R : [[n/r]]^2 \rightarrow Bool$
3: $\quad$ **return** $(i < j$ and $d_n(i, j) \equiv 0 \pmod 2)$ or $(i > j$ and $d_n(i, j) \equiv 1 \pmod 2)$
4: **for all** $i \in [[n/r]]$ **do**
5: $\quad P_i \leftarrow \{(i-1) \cdot r + 1, \ldots, i \cdot r\}$
6: $\quad A \leftarrow \varnothing \triangleright A : \mathcal{P}(P_i) \rightarrow \mathcal{P}(V)$, *implemented as an array of linked lists of size* $2^{|P_i|} = O\left(\frac{n}{\lg^2 n}\right)$
7: $\quad$ **for all** $v \in V$ **do**
8: $\qquad$ **if** $R(\lceil v/r \rceil, i)$ **then**
9: $\qquad\quad S \leftarrow \varnothing \triangleright S : \mathcal{P}(P_i)$
10: $\qquad\quad$ **for all** $u \in P_i$ **do**
11: $\qquad\qquad$ **if** $\{u, v\} \in E$ **then**
12: $\qquad\qquad\quad S \leftarrow S \cup \{u\}$
13: $\qquad\quad A(S) \leftarrow A(S) \cup \{v\} \triangleright$ *We think of S as a binary string that indexes A*
14: $\quad$ **for all** $S \in \mathcal{P}(P_i)$ **do**
15: $\qquad$ **if** $|S| > 0$ and $|A(S)| > 0$ **then**
16: $\qquad\quad$ **Emit** $(S, A(S))$
17: $\quad$ **for all** $\{u, v\} \in \binom{P_i}{2}$ **do**
18: $\qquad$ **if** $\{u, v\} \in E$ **then**
19: $\qquad\quad$ **Emit** $(\{u\}, \{v\})$

Proof. Let $\mathcal{G}_n$ be the set of graphs with vertex set $[n]$. We will show there exists an injective function $f : \mathcal{G}_n \rightarrow \{0, 1\}^\star$ such that $|f(G)| \leq \mathsf{BC}(G) \cdot (\lg n + O(1))$ for every $G \in \mathcal{G}_n$, and thus $|s| \leq \mathsf{BC}(n) \cdot (\lg n + O(1))$ for every $s \in \mathrm{range}(f)$. Note that the theorem follows from this: there are at most $2^{\mathsf{BC}(n) \cdot (\lg n + O(1))}$ binary strings of length at most $\mathsf{BC}(n) \cdot (\lg n + O(1))$, and thus $|\mathrm{range}(f)| \leq 2^{\mathsf{BC}(n) \cdot (\lg n + O(1))}$, but by the injectivity of f we deduce $2^{\binom{n}{2}} = |\mathrm{domain}(f)| \leq |\mathrm{range}(f)| \leq 2^{\mathsf{BC}(n) \cdot (\lg n + O(1))}$, from where the result follows by taking logarithms.

It only remains to show that such an f exists. Indeed, identify the vertices of an n-vertex graph G with binary strings of length $\lceil \lg n \rceil$, and let $\circ$ denote concatenation. Then, given a biclique (X, Y) in G with $n_1 := |X|$ and $n_2 := |Y|$, let $v_1, \ldots, v_{n_1}$ be the vertices of X in lexicographic order, and let $w_1, \ldots, w_{n_2}$ be the vertices of Y in lexicographic order. Then, we can represent the biclique (X, Y) with the binary string $\mathrm{enc}(X, Y) := 0^{n_1} \circ 1 \circ 0^{n_2} \circ 1 \circ v_1 \circ \cdots \circ v_{n_1} \circ w_1 \circ \cdots \circ w_{n_2}$ of length $(n_1 + 1) + (n_2 + 1) + (n_1 + n_2) \lceil \lg n \rceil = (n_1 + n_2)(\lg n + O(1))$. Now, given a biclique cover $\mathcal{B} = \{(X_1, Y_1), \ldots, (X_k, Y_k)\}$, we can represent it with the binary string $\mathrm{enc}(X_1, Y_1) \circ \mathrm{enc}(X_2, Y_2) \circ \cdots \circ \mathrm{enc}(X_k, Y_k)$ of length $w(\mathcal{B}) \cdot (\lg n + O(1))$, and let $g(\mathcal{B})$ be one of these strings chosen arbitrarily. Let $f(G) := g(\mathcal{B})$, where $\mathcal{B}$ is a biclique cover of weight $\mathsf{BC}(G)$ chosen arbitrarily. Then, given an n-vertex graph G, $f(G)$ is a binary string of length at most $\mathsf{BC}(G)(\lg n + O(1))$. Moreover, f is injective on the set of n-vertex graphs, since a biclique cover uniquely specifies an n-vertex graph and the encoding g is injective. This concludes the proof. □

Theorem 4 follows immediately from Theorems 11 and 12. The lower bound proof can be extended to almost all graphs, i.e., with high probability in the $G(n, 1/2)$ model, answering another question of Chung, Erdős, and Spencer. We generalize this to the $G(n, p)$ model for other values of p in Section 8.

It is worth commenting that our upper bound proof is similar to that of [Lup56], and also that

of [CLT15], but we get a $\frac{1}{2}$ factor through the almost-regular tournament idea, whereas [CLT15] orient each edge, which ends up being a worse choice. In fact, the almost-regularity idea can be seen as a very basic form of *equitability*, a concept we will develop much more powerfully when generalizing to hypergraphs.

Furthermore, the previous analysis can easily be extended to directed graphs. We do not allow directed graphs to have loops, since a loop cannot be covered by a directed biclique.

Theorem 13. *Every directed graph $G = (V, E)$ can be partitioned into directed bicliques such that every vertex is contained in at most $(1 + o(1))\frac{n}{\lg n}$ of the directed bicliques, and this is asymptotically optimal.*

Proof. The lower bound follows directly from recreating the proof of Theorem 12 but noting that there are 2^{n^2-n} directed graphs on n vertices. For the upper bound, the construction is very similar to the proof of Theorem 11, albeit even simpler. We again let $r := \lfloor \lg n - 2 \lg \lg n \rfloor$ and partition $V(G)$ into $\lceil n/r \rceil$ parts $P_1, \dots, P_{\lceil n/r \rceil}$ of size at most r each. Given a vertex $v \in V(G)$, let $g(v)$ be the index of the part P_i containing v. Now, for each part P_i, and every subset $S \subseteq P_i$, we define $A(S) := \{v \in V : N(v) \cap P_i = S\}$. Consider the set of directed bicliques $\mathcal{C} := \{(S, A(S)) \mid S \subseteq P_i, i \in [\lceil n/r \rceil]\}$ Given $(u, v) \in E(G)$, let $S = N(u) \cap P_{g(v)}$. Then, it is not hard to see that (u, v) is covered uniquely by $(A(S), S)$, so $\mathcal{C}$ is a directed biclique partition of G. The proof that each vertex is contained in at most $(1 + o(1))\frac{n}{\lg n}$ directed bicliques from $\mathcal{C}$ is very similar to the proof of Theorem 11. $\square$

3 Generalization to hypergraphs

In this section we prove Theorem 5. For the sake of exposition, we begin with a weaker result that only bounds $\mathsf{MC}_d(n)$ and $\mathsf{MP}_d(n)$, but has a much simpler proof.

3.1 Optimal Chung–Erdős–Spencer for hypergraphs

Theorem 14. *For every $d \geq 2$, we have $\left(\frac{1}{d!} - o_d(1)\right)\frac{n^d}{\lg n} \leq \mathsf{MC}_d(n) \leq \mathsf{MP}_d(n) \leq \left(\frac{1}{d!} + o_d(1)\right)\frac{n^d}{\lg n}$.*

To see the lower bound, note that the proof of Theorem 12 naturally generalizes to d-graphs, with the only essential difference being that there are $2^{\binom{n}{d}}$ different d-graphs on n vertices. For the proof of the upper bound we need the following lemma, which will enable an induction on d.

Lemma 15. *Let f, g be non-decreasing functions and $d \geq 2$. Suppose that each n-vertex d-uniform hypergraph can be partitioned into at most $f(n)$ many d-cliques with total weight at most $g(n)$. Then every n-vertex $(d + 1)$-graph can be partitioned into at most $n \cdot f(n)$ many $(d + 1)$-cliques with total weight at most $\sum_{i=0}^{n-1}(f(i) + g(i))$.*

Proof. Let H be an n-vertex $(d + 1)$-uniform hypergraph, and order its vertices arbitrarily as $v_1, \dots, v_n$. For each $i \in [n]$, let $H_i \subseteq H$ be the $(d + 1)$-uniform hypergraph defined by $V(H_i) = \{v_i, \dots, v_n\}$ and $E(H_i) = \{e \in E(H) : e \subseteq \{v_i, \dots, v_n\}$ and $v_i \in e\}$. Note that $H_1, \dots, H_n$ partition $E(H)$. For each $i \in [n]$, let H_i' be the d-uniform hypergraph defined by $V(H_i') = \{v_{i+1}, \dots, v_n\}$ and $E(H_i') = \{e : e \cup \{v_i\} \in E(H_i)\}$. By assumption, each H_i' has a partition $\mathcal{C}_i'$ into at most $f(n-i) \leq f(n)$ d-cliques with total weight at most $g(n-i)$. Each d-clique $(A_1, \dots, A_d)$ in H_i' can be extended to $(d + 1)$-clique $(\{v_i\}, A_1, \dots, A_d)$ in H_i. Applying this to each d-clique in $\mathcal{C}_i'$, we obtain a family $\mathcal{C}_i$, which partitions H_i, consisting of at most $f(n)$ d-cliques, and with total weight at most $f(n-i) + g(n-i)$. Then, the family $\mathcal{F}_1 \cup \dots \cup \mathcal{F}_n$ partitions $E(H)$ and satisfies the required properties. $\square$

9

Proof of Theorem 14. We prove, by induction on $d \geqslant 2$, that n-vertex d-graphs can be partitioned into at most $O_d(n^d/\lg^3 n)$ many d-cliques with total weight at most $(1/d! + o_d(1))n^d/\lg n$. We know this is true for $d = 2$, by Theorem 11. Assuming it is true for d, we prove it for $d + 1$. Using Lemma 15 with $f(n) = O_d(n^d/\lg^3 n)$ and $g(n) = (1/d! + o_d(1))n^d/\lg n$, we get that each $(d+1)$-graph can be partitioned into at most $nf(n) = O_d(n^{d+1}/\lg^3 n)$ many $(d+1)$-cliques, and the total weight is at most

$$\sum_{i=0}^{n-1}(f(i) + g(i)) = \left(\frac{1}{d!} + o_d(1)\right)\sum_{i=0}^{n-1}\frac{i^d}{\lg i},$$

where we used that $f(n) = o_d(g(n))$. To analyze the last sum, we write it as $S_1 + S_2$, where S_1 corresponds to the first $t := \lfloor n/\lg n \rfloor$ terms, and S_2 to the remaining terms. We clearly have $S_1 \leqslant \sum_{i=0}^{t} t^d \leqslant t^{d+1} = o_d(n^{d+1}/\lg n)$. On the other hand, for $t \leqslant i \leqslant n$ we have $\lg i = (1 + o_d(1))\lg n$, and therefore $S_2 \leqslant \frac{(1+o_d(1))}{d!\lg n}\sum_{i=t}^{n} i^d$. Hence, to conclude it suffices to observe that $\sum_{i=1}^{n} i^d \leqslant (1 + o_d(1))\frac{1}{d+1}n^{d+1}$. A quick argument follows by estimating the sum using an integral: we have $i^d \leqslant \int_i^{i+1} x^d \mathrm{d}x$ by monotonicity of $x \mapsto x^d$ for $x \in [1, \infty)$, so

$$\sum_{i=1}^{n} i^d \leqslant \int_1^{n+1} x^d \mathrm{d}x = \frac{x^{d+1}}{d+1}\Big|_1^{n+1} \leqslant \frac{(n+1)^{d+1}}{d+1} = \frac{n^{d+1}}{d+1}\cdot(1+\tfrac{1}{n})^{d+1} \leqslant \frac{n^{d+1}}{d+1}e^{(d+1)/n},$$

we conclude by observing that $e^{(d+1)/n} = 1 + o_d(1)$. $\qquad\square$

The above argument gives a deterministic $O(n^d)$-time algorithm to find a d-clique partition of weight at most $\left(\frac{1}{d!} + o_d(1)\right)\frac{n^d}{\lg n}$: in the $d = 2$ case we use Algorithm 1. For $d > 2$ we call n times the $O(n^{d-1})$-time algorithm for $(d-1)$-graphs, as done in the proof of Lemma 15.

3.2 Optimal Erdős–Pyber for hypergraphs

To complete the proof of Theorem 5 we need to show the bound $\mathsf{LMP}_d(n) \leqslant \left(\frac{1}{d!} + o_d(1)\right)\frac{n^{d-1}}{\lg n}$ for each $d \geqslant 2$. This is the content of Theorem 18, for which we need some preparations.

First, let us analyze why the proof of Theorem 14 in the preceding subsection does not allow for a good upper bound on the number of d-cliques an arbitrary vertex v participates in. Note that, if we unfold the induction from Lemma 15, the constructed d-cliques have exactly two parts of size larger than 1, and $d - 2$ parts of size 1. The construction can be thought of as creating, for each subset $S := \{s_1, \ldots, s_{d-2}\} \subset V(H)$ of $d - 2$ vertices, a "link graph" G_S whose edges are the pairs $\{v_i, v_j\}$ such that $\{v_i, v_j\} \cup S \in E(H)$. Then, each of these link graphs G_S can be partitioned into bicliques (L_i, R_i) using Theorem 11, resulting in d-cliques $(\{s_1\}, \ldots, \{s_{d-2}\}, L_i, R_i)$. For these to be an actual partition of $E(H)$, it is required that each hyperedge gets placed into exactly one link graph, which Lemma 15 ensures by using a fixed ordering of the vertices and placing a hyperedge in the link graph corresponding to its first $d - 2$ vertices in the ordering. Note that the load (i.e., number of d-cliques) of every vertex v is thus coming from two sources. First, there are the d-cliques in link graphs G_S where $v \in S$, which for a fixed S are exactly the bicliques into which G_S gets partitioned by Theorem 11, of which there are $O(n^2/\lg^3 n)$. On the other hand, there are the d-cliques in link graphs for which $v \notin S$, which for a fixed S are counted by the load of v in the biclique partition of G_S; this is at most $(1 + o(1))\frac{n}{2\lg n}$. The former are not many: there are at most $\binom{n-1}{d-3} = O(n^{d-3})$ possibilities for $S \ni v$, and therefore their associated load is at most
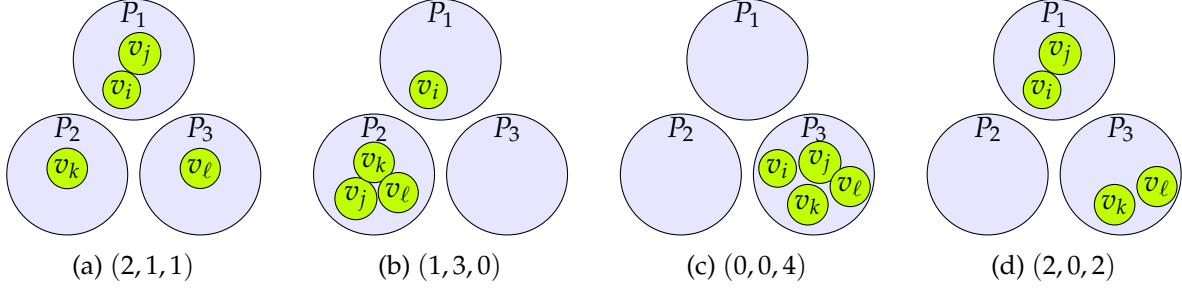
10

Figure 4: The 4 distributions for $d = 4$ up to permutation.

$O(n^{d-3} \cdot n^2 / \lg^3 n) = O(n^{d-1} / \lg^3 n) = o_d(\frac{n^{d-1}}{\lg n})$. The bottleneck is when $v \notin S$, since then if v is the last vertex in the fixed ordering, there are about $\binom{n-1}{d-2}$ choices for S, and thus the upper bound we get on the load of v is at most

$$\binom{n-1}{d-2} \cdot (1 + o(1)) \frac{n}{2 \lg n} \leqslant (1 + o(1)) \frac{n^{d-1}}{2(d-2)! \lg n},$$

which is not good enough; that bound is only a factor 2 better than the upper bound of [CLT15], and this factor comes only from our improvement for graphs. The cause of this issue is that the strategy used to decide in which link graph to place each hyperedge is very "inequitable"; vertices that come late in the ordering get too many, and the ones that come early get too few. This motivates the design of equitable strategies for placing hyperedges into link graphs.

Now, before proving Theorem 18 for arbitrary d, let us consider a particular case such as $d = 4$ to motivate the conceptual design of our proof. In this case, the main question is *in which link graph should we place a given hyperedge $e := \{v_i, v_j, v_k, v_\ell\}$?* Suppose we first partition $V(H)$ into $d - 1 = 3$ parts P_1, P_2, P_3 of roughly equal size. Then, a "distribution" tuple $(|P_1 \cap e|, |P_2 \cap e|, |P_3 \cap e|)$ will capture how e is distributed along the parts, as illustrated in Figure 4. We will construct link graphs whose vertices are completely contained within a part, since this way each link graph will have roughly $n/(d-1) = n/3$ vertices instead of n. In consequence, the case presented in Figure 4a has only one possible choice: to place the edge $\{v_i, v_j\}$ in the link graph $G_{\{v_k, v_\ell\}}$. To handle the case presented in Figure 4b, however, we will consider a 3-uniform link hypergraph $H_{\{v_i\}}$, in which the edge $\{v_j, v_k, v_\ell\}$ will be placed, and then recursion with $d = 3$ over $H_{\{v_i\}}$ will take care of splitting hyperedges such as $\{v_j, v_k, v_\ell\}$ in such a way that each of those 3 vertices ends up with the same load. Next, the case presented in Figure 4c will be handled by recursion with $n' := |P_3| \approx n/3$, which will further partition P_3. The last case, in Figure 4d, is the most interesting since for the first time there is a choice: shall we place $\{v_i, v_j\}$ into $G_{\{v_k, v_\ell\}}$ (thus loading P_1 more) or $\{v_k, v_\ell\}$ into $G_{\{v_i, v_j\}}$ (thus loading P_3 more)? The important aspect here is that we need to make a choice that is not biased towards any particular part, so we can choose to e.g., load P_3 more for distribution $(2, 0, 2)$, as long as P_1 gets more load from $(2, 2, 0)$ and P_2 get more from $(0, 2, 2)$.

Note that, a key idea used here is that the decision of what to do with e is purely based on its distribution tuple, and not on the precise identity of its vertices. Thus, the number of link graphs in which edge $\{v_i, v_j\}$ is placed by a distribution tuple $(2, 1, 1)$, as in Figure 4a, is simply $|P_2| \cdot |P_3| \approx n^2/9$, and each of those incurs in load roughly $|P_1|/(2 \lg |P_1|) \approx n/(6 \lg n)$ by Theorem 11, thus resulting in load $n^3/(54 \lg n)$ associated to tuples $\{(2, 1, 1), (1, 2, 1), (1, 1, 2)\}$ for each vertex. In turn, when considering a distribution tuple like $(1, 3, 0)$, the vertices in P_2 will be part of $|P_1| \approx n/3$ link hypergraphs like $H_{\{v_i\}}$, and by inductive hypothesis, this will result in a load of roughly $\frac{n}{3} \cdot \frac{(n/3)^2}{3! \lg n} = n^3/(162 \lg n)$, which after accounting for the symmetric

11

tuple $(0, 3, 1)$, which also puts load on P_2, this case results in $n^3/(81 \lg n)$ per vertex. When considering a distribution tuple like $(0, 0, 4)$, as in Figure 4c, by inductive hypothesis this will incur in load roughly $(n/3)^3/(4! \lg n) = n^3/(648 \lg n)$. Finally, when considering distributions in $\{(0, 2, 2), (2, 0, 2), (2, 2, 0)\}$, each vertex v only needs to consider the single one that is loaded to its part; this yields a load of roughly $\binom{n/3}{2} \cdot \frac{n/3}{2 \lg n} \approx n^3/(108 \lg n)$. Aggregating the different cases, the load of every vertex is at most

$$(1 + o(1)) \frac{n^3}{\lg n} \left(\frac{1}{54} + \frac{1}{81} + \frac{1}{648} + \frac{1}{108} \right) = \left(\frac{1}{4!} + o(1) \right) \frac{n^3}{\lg n},$$

as desired.

It is not obvious, however, how to generalize this reasoning for arbitrary d: how should we proceed with distributions like $(2, 2, 3, 3, 0, 0, 0, 0, 0)$ or $(2, 2, 2, 2, 1, 1, 0, 0, 0)$ for $d = 10$? This is where the notion of equitability comes in. Every potential hyperedge (i.e., subset of d vertices) corresponds to a distribution $\vec{x} = (x_1, \ldots, x_{d-1})$ according to how many vertices are in each part, and given such a distribution, we must make some choice of which part is load-bearing, which we call $f(\vec{x}) \in [d-1]$. Given a random potential hyperedge, we want the probability that each part $i \in [d-1]$ is load-bearing in the corresponding distribution to be roughly $1/(d-1)$; more specifically, we want

$$\sum_{(x_1, \ldots, x_{d-1}) \in f^{-1}(i)} \frac{\binom{d}{x_1, \ldots, x_{d-1}}}{(d-1)^d} = \frac{1}{d-1}$$

for each $i \in [d-1]$. This motivates the following definition.

Definition 16. Let $d \geqslant 2$ be an integer. A *d-distribution* is a tuple $(x_1, \ldots, x_{d-1}) \in \mathbb{N}^{d-1}$ such that $x_1 + \ldots + x_{d-1} = d$. Let $\mathcal{S}_d$ be the set of d-distributions. A *selection strategy* is a function $f : \mathcal{S}_d \to [d-1]$ such that $f(x_1, \ldots, x_{d-1}) = i$ for some i such that $x_i \geqslant 2$. Then, we say a selection strategy is *equitable* if for all $i, j \in [d-1]$,

$$\sum_{(x_1, \ldots, x_{d-1}) \in f^{-1}(i)} \binom{d}{x_1, \ldots, x_{d-1}} = \sum_{(x_1, \ldots, x_{d-1}) \in f^{-1}(j)} \binom{d}{x_1, \ldots, x_{d-1}}.$$

Lemma 17. *For every $d \geqslant 2$, there exists an equitable selection strategy.*

Proof. Fix an integer $d \geqslant 2$. Given a d-distribution $(x_1, \ldots, x_{d-1})$, let $[(x_1, \ldots, x_{d-1})]$ be its set of cyclic rotations; that is, $[(x_1, \ldots, x_{d-1})] = \{(x_i, \ldots, x_{d-1}, x_1, \ldots, x_{i-1}) : i \in [d-1]\}$. Note that $[\cdot]$ partitions $\mathcal{S}_d$ into equivalence classes.

First, we claim that for every d-distribution $(x_1, \ldots, x_{d-1})$, we have $|[(x_1, \ldots, x_{d-1})]| = d - 1$. By a standard result from combinatorics on words (see, e.g., [Lot02, Section 1.2.1]), this is equivalent to the claim that every d-distribution is *primitive*, meaning that it is not of the form $(x_1, \ldots, x_j)^n$ for some $j \geqslant 1$ and $n \geqslant 2$, where the exponent means repeated concatenation. And, indeed, if $(x_1, \ldots, x_{d-1}) = (x_1, \ldots, x_j)^n$, then $n \cdot \left(\sum_{i=1}^{j} x_i \right) = \sum_{i=1}^{d-1} x_i = d$. Since we also have $n \cdot j = d - 1$, we have that n divides both d and $d - 1$, so $n = 1$, proving the claim.

Now, we construct our selection strategy f as follows. For each equivalence class $[\vec{x}] \in \mathcal{S}_d/[\cdot]$, pick a representative $(x_1, \ldots, x_{d-1}) \in [\vec{x}]$ such that $x_1 \geqslant 2$. Then, let $f(x_1, \ldots, x_{d-1}) = 1$ and $f(x_i, \ldots, x_{d-1}, x_1, \ldots, x_{i-1}) = 1 + (d - i)$ for all $i \in [2, d-1]$. We have that the $f(\vec{x})$-th entry of $\vec{x}$ equals $x_1 \geqslant 2$ for each $\vec{x} \in [(x_1, \ldots, x_{d-1})]$, so f is a selection strategy. Furthermore, for each equivalence class $[\vec{x}] \in \mathcal{S}_d/[\cdot]$ and $i \in [d-1]$, we have $|f^{-1}(i) \cap [\vec{x}]| = 1$. Therefore, the selection strategy is equitable. $\square$

12

Theorem 18. *We have $LMP_d(n) \leqslant \left(\frac{1}{d!} + o_d(1)\right)\frac{n^{d-1}}{\lg n}$, and there is a d-clique partition realizing this bound in which the number of d-cliques is $O_d(n^d / \lg^3(n))$. Furthermore, a d-clique partition realizing this upper bound can be constructed deterministically in time $O(n^d / d!)$.*

Proof. More precisely, we prove that for every $d \geqslant 2$, there are functions $r_d(n) = o_d(n^{d-1}/\lg n)$ and $h_d(n) = O_d(n^d/\lg^3(n))$ such that every d-uniform hypergraph H on n vertices can be written as an edge-disjoint union of d-cliques such that each vertex is in at most

$$g_d(n) := \frac{1}{d!} \cdot \frac{n^{d-1}}{\lg n} + r_d(n) = \left(\frac{1}{d!} + o_d(1)\right)\frac{n^{d-1}}{\lg n} \tag{1}$$

d-cliques, where the number of d-cliques is at most $h_d(n)$. Concretely, we will take

$$r_d(n) = \frac{4^d}{d!} \cdot 2^{d2^{100d^2}} n^{d-1}\frac{\lg\lg n}{\lg^2 n}, \quad h_d(n) = 2^{d2^{100d^2}}\frac{n^d}{\lg^3 n},$$

which essentially amounts to assuming that n is much larger than d, since for $n \leqslant 2^{2^{100d^2}}$ we have

$$n^d \leqslant \left(2^{2^{100d^2}}\right)^d = 2^{d2^{100d^2}} \leqslant \min\{r_d(n), h_d(n)\},$$

and thus the statement holds trivially.

The proof of the theorem proceeds by strong induction on d. The base case $d = 2$ corresponds to Theorem 11, and for the inductive case, with $d \geqslant 3$, the proof is by strong induction on n.

First, we partition the vertices of H into parts $P_1, \ldots, P_{d-1}$ such that $||P_i| - |P_j|| \leqslant 1$ for all $i, j \in [d-1]$. Let f be an equitable selection strategy on $\mathcal{S}_d$, which exists by Lemma 17. For each d-distribution $\vec{x} = (x_1, \ldots, x_{d-1}) \in \mathcal{S}_d$, we create a set of d-clique covers as follows. Let $i = f(\vec{x})$, and

$$\mathcal{A}_{\vec{x}} := \{S \subseteq V(H)\backslash P_i : |S \cap P_j| = x_j \text{ for all } j \in [d-1]\backslash\{i\}\}.$$

Note that $\mathcal{A}_{\vec{x}}$ consists of sets of vertices of size $k - x_i$ each. For each $S \in \mathcal{A}_{\vec{x}}$, let $H_{\vec{x},S}$ be the x_i-uniform hypergraph defined by $V(H_{\vec{x},S})L = P_i$ and $E(H_{\vec{x},S}) = \{T : T \subseteq P_i, S \cup T \in E(H)\}$. Let $\mathcal{C}_{\vec{x},S}$ be an x_i-clique partition of $H_{\vec{x},S}$ such that each vertex of P_i is in at most $g_{x_i}(|P_i|)$ many x_i-cliques and $|\mathcal{C}_{\vec{x},S}| \leqslant h_{x_i}(|P_i|)$, which exists by induction. Given an x_i-clique $(A_1, \ldots, A_{x_i})$ in $\mathcal{C}_{\vec{x},S}$, we can extend it to a d-clique $S \oplus \{A_1, \ldots, A_{x_i}\} \subseteq H$, which is defined as the d-clique in H whose parts are $\{\{v\} : v \in S\} \cup \{A_1, \ldots, A_{x_i}\}$. Let

$$\mathcal{C}'_{\vec{x},S} = \{S \oplus \{A_1, \ldots, A_{x_i}\} : (A_1, \ldots, A_{x_i}) \in \mathcal{C}_{\vec{x},S}\} \quad \text{and} \quad \mathcal{C} := \bigcup_{\substack{\vec{x} \in \mathcal{S}_d \\ S \in \mathcal{A}_{\vec{x}}}} \mathcal{C}'_{\vec{x},S}.$$

First, we argue that $\mathcal{C}$ is a d-clique partition of G. Let $e \in E(H)$, and let $\vec{x} = (x_1, \ldots, x_{d-1}) \in \mathcal{S}_d$ be such that $e \cap P_i = x_i$ for all $i \in [d-1]$. Let $i = f(\vec{x})$. Then, there is a unique $S \in \mathcal{A}_{\vec{x}}$ such that $S \subseteq e$, and e is uniquely covered by a d-clique in $\mathcal{C}'_{\vec{x},S}$. On the other hand, d is not covered by any d-clique in $\mathcal{C}'_{\vec{x}',S'}$ for any $(\vec{x}', S') \neq (\vec{x}, S)$.

Next, we argue that each vertex is in at most $g_d(n)$ many d-cliques from $\mathcal{C}$. Fix a vertex $v \in V(H)$, and suppose that $v \in P_i$. There are two types of d-cliques from $\mathcal{C}$ that we consider:

1. **(Core)** These are the d-cliques in some $\mathcal{C}'_{\vec{x},S}$ for which $f(\vec{x}) = i$.

13

2. (**Auxiliary**) These are the d-cliques in some $\mathcal{C}'_{\vec{x},S}$ for which $f(\vec{x}) \neq i$.

We now bound these separately, for which we will first need some mildly technical conditions on the functions r_d and h_d:

Claim 19. The functions r_d and h_d satisfy the following properties.

1. $r_d(n) \geqslant 4\frac{n}{d}r_{d-1}(n)$, for every n, d.

2. $r_d(n+1) \geqslant r_d(n)$ and $h_d(n+1) \geqslant h_d(n)$, for every n, d.

3. $r_d(\lceil n/(d-1)\rceil) \leqslant 1.1 \frac{r_d(n)}{(d-1)^{d-1}}$, for every $n \geqslant 2^{2^{100d^2}}$.

The proof of Claim 19 is deferred to Appendix A.

Claim 20. The number of core d-cliques containing v is at most $\frac{1}{d!} \cdot \frac{n^{d-1}}{\lg n} + \frac{2}{3}r_d(n)$.

Proof of Claim 20. By the inductive hypothesis (using induction on d if $x_i < d$, and on n if $x_i = d$), the number of core d-cliques containing v is at most

$$\sum_{(x_1,\ldots,x_{d-1})\in f^{-1}(i)} \left(\prod_{j\in[d-1]\setminus\{i\}} \binom{|P_j|}{x_j}\right) \cdot g_{x_i}(|P_i|)$$

$$\leqslant \sum_{(x_1,\ldots,x_{d-1})\in f^{-1}(i)} \left(\prod_{j\in[d-1]\setminus\{i\}} \frac{|P_j|^{x_j}}{x_j!}\right) \cdot \left(\frac{1}{x_i!} \cdot \frac{|P_i|^{x_i-1}}{\lg |P_i|} + r_{x_i}(|P_i|)\right) := \dagger.$$

Since each part P_k has size at most $\lceil n/(d-1)\rceil$, and $\sum_{k=1}^{d-1} x_k = d$, we have

$$\dagger \leqslant \sum_{(x_1,\ldots,x_{d-1})\in f^{-1}(i)} \left(\frac{1}{x_1!\cdot\ldots\cdot x_{d-1}!}\right) \cdot \left\lceil\frac{n}{d-1}\right\rceil^{d-1} \cdot \left(\frac{1}{\lg |P_i|} + \frac{x_i!}{|P_i|^{x_i-1}}r_{x_i}(|P_i|)\right).$$

We now split the last summation according to whether $x_i = d$ or not. We thus have

$$\dagger \leqslant \left\lceil\frac{n}{d-1}\right\rceil^{d-1} \cdot \frac{1}{d!}\left(\frac{1}{\lg |P_i|} + \frac{d!}{|P_i|^{d-1}}r_d(|P_i|)\right)$$

$$+ \left\lceil\frac{n}{d-1}\right\rceil^{d-1} \cdot \sum_{\substack{(x_1,\ldots,x_{d-1})\in f^{-1}(i)\\ x_i \neq d}} \left(\frac{1}{x_1!\cdot\ldots\cdot x_{d-1}!}\right) \cdot \left(\frac{1}{\lg |P_i|} + \frac{(d-1)!}{|P_i|^{d-2}}r_{d-1}(|P_i|)\right).$$

Now we note that

$$\sum_{(x_1,\ldots,x_{d-1})\in\mathcal{S}_d} \left(\frac{d!}{x_1!\cdot\ldots\cdot x_{d-1}!}\right) = (d-1)^d, \tag{2}$$

since both sides count the number of distinct strings of length d over an alphabet of size $d-1$. Then, by equitability of f, we have that

$$\sum_{(x_1,\ldots,x_{d-1})\in f^{-1}(i)} \left(\frac{d!}{x_1!\cdot\ldots\cdot x_{d-1}!}\right) = \frac{1}{d-1} \sum_{(x_1,\ldots,x_{d-1})\in\mathcal{S}_d} \left(\frac{d!}{x_1!\cdot\ldots\cdot x_{d-1}!}\right) = (d-1)^{d-1}.$$

14

Also, there is a unique d-distribution $\vec{x} = (x_1, \ldots, x_d)$ such that $x_i = d$ (and $x_j = 0$ for all $j \neq i$), and for such $\vec{x}$ we must have $f(\vec{x}) = i$. Therefore,

$$\sum_{\substack{(x_1,\ldots,x_{d-1}) \in f^{-1}(i) \\ x_i \neq d}} \left(\frac{1}{x_1! \cdot \ldots \cdot x_{d-1}!} \right) = \frac{(d-1)^{d-1} - 1}{d!}. \tag{3}$$

Using Equation (3) in our last inequality regarding †, and introducing notation $w_d(x) := \frac{d!}{x^{d-1}} r_d(x)$, we have

$$\dagger \leqslant \frac{[n/(d-1)]^{d-1}}{d!} \cdot \left[\frac{(d-1)^{d-1}}{\lg |P_i|} + w_d(|P_i|) + ((d-1)^{d-1} - 1) w_{d-1}(|P_i|) \right].$$

As $r_d(x) \geqslant 4\frac{x}{d} r_{d-1}(x)$ by Claim 19, we have that $w_{d-1}(x) \leqslant w_d(x)/4$. Using this, we get

$$\dagger \leqslant \frac{[n/(d-1)]^{d-1}}{d!} \cdot \left(\frac{(d-1)^{d-1}}{\lg |P_i|} + w_d(|P_i|) \left[\frac{(d-1)^{d-1}}{4} + 1 \right] \right)$$

$$\leqslant \frac{[n/(d-1)]^{d-1}}{d!} \cdot \left(\frac{(d-1)^{d-1}}{\lg(\lfloor n/(d-1) \rfloor)} + \frac{d!}{|P_i|^{d-1}} r_d(|P_i|) \left[\frac{(d-1)^{d-1}}{4} + 1 \right] \right).$$

As $\lceil n/(d-1) \rceil \leqslant (n+d-2)/(d-1)$, $|P_i| \geqslant \lfloor n/(d-1) \rfloor \geqslant \frac{n-d+2}{d-1}$, and r_d is an increasing function, we now have

$$\dagger \leqslant \frac{(n+d-2)^{d-1}}{d! \lg(\lfloor n/(d-1) \rfloor)} + \left(\frac{n+d-2}{n-d+2} \right)^{d-1} r_d\left(\left\lceil \frac{n}{d-1} \right\rceil \right) \left(\frac{(d-1)^{d-1}}{4} + 1 \right),$$

and since $n \geqslant 2^{2^{200d^2}}$, it is easy to see that $\left(\frac{n+d-2}{n-d+2} \right)^{d-1} \leqslant 1.1$, from where

$$\dagger \leqslant \frac{(n+d-2)^{d-1}}{d! \lg(\lfloor n/(d-1) \rfloor)} + 1.1 \cdot 1.1 r_d(n) \left(\frac{1}{4} + \frac{1}{(d-1)^{d-1}} \right)$$

$$\leqslant \frac{(n+d-2)^{d-1}}{d! \lg(\lfloor n/(d-1) \rfloor)} + 1.1 \cdot 1.1 r_d(n) \left(\frac{1}{4} + \frac{1}{4} \right) \qquad \text{(As } d \geqslant 3\text{)}$$

$$\leqslant \frac{(n+d-2)^{d-1}}{d! \lg(\lfloor n/(d-1) \rfloor)} + 0.61 r_d(n).$$

To conclude, it suffices to prove that $\frac{(n+d-2)^{d-1}}{d! \lg(\lfloor n/(d-1) \rfloor)} \leqslant \frac{n^{d-1}}{d! \lg n} + 0.05 r_d(n)$. First, use that $(n + d - 2)^{d-1} \leqslant (n+d)^{d-1} \leqslant n^{d-1} + d^{d+1} \cdot n^{d-2}$, where the last inequality follows from upper bounding all but the first term in the binomial expansion of $(n+d)^{d-1}$. It is then easy to see that $d^{d+1} \cdot n^{d-2} \leqslant \frac{1}{100} r_d(n)$. Thus, it only remains to prove that $\frac{n^{d-1}}{d! \lg(\lfloor n/(d-1) \rfloor)} \leqslant \frac{n^{d-1}}{d! \lg n} + 0.04 r_d(n)$, which follows from writing

$$\frac{n^{d-1}}{d! \lg(\lfloor n/(d-1) \rfloor)} \leqslant \frac{n^{d-1}}{d! (\lg(n-d+2) - \lg(d-1))} = \frac{n^{d-1}}{d! \lg n} \cdot \frac{\lg n}{\lg(n-d+2) - \lg(d-1)}$$

$$= \frac{n^{d-1}}{d! \lg n} \cdot \left(1 + \frac{\lg n - \lg(n-d+2) + \lg(d-1)}{\lg(n-d+2) - \lg(d-1)} \right)$$

$$\leqslant \frac{n^{d-1}}{d! \lg n} \cdot \left(1 + \frac{3 \lg d}{\lg n} \right) = \frac{n^{d-1}}{d! \lg n} + \frac{3 \lg d \cdot n^{d-1}}{d! \lg^2 n}$$

$$\leqslant \frac{n^{d-1}}{d! \lg n} + 0.01 r_d(n). \qquad \text{(End of proof of Claim 20)} \quad \blacksquare$$

Claim 21. The number of auxiliary d-cliques containing v is at most $\frac{1}{3}r_d(n)$.

The proof of Claim 21 is similar to the previous one, and thus deferred to Appendix A. From these two claims it immediately follows that v is contained in at most $g_d(n)$ different d-cliques. We also need, in order to preserve our inductive hypothesis, to establish the following claim, whose proof is also in Appendix A.

Claim 22. We have $|\mathcal{C}| \leqslant h_d(n)$.

Finally, it remains to show that the construction can be deterministically carried out in time $O(n^d/d!)$. Let $T_d(n)$ be the time complexity of the above construction, and we want to show that there is an absolute constant C such that $T_d(n) \leqslant C \cdot n^d/d!$. The proof is again by a strong double induction, first on d and then on n. We have

$$
\begin{aligned}
T_d(n) &\leqslant \sum_{\vec{x}\in\mathcal{S}_d} \left(\prod_{i\in[d-1]\setminus\{f(\vec{x})\}} \binom{|P_i|}{x_i} \right) T_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|) \\
&\leqslant \sum_{\vec{x}\in\mathcal{S}_d} \left(\prod_{i\in[d-1]\setminus\{f(\vec{x})\}} \frac{|P_i|^{x_i}}{x_i!} \right) T_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|) \\
&\leqslant C \cdot \sum_{\vec{x}\in\mathcal{S}_d} \prod_{i\in[d-1]} \frac{|P_i|^{x_i}}{x_i!} \qquad\qquad\qquad \text{(by induction)} \\
&= \frac{C}{d!} \sum_{\vec{x}\in\mathcal{S}_d} \left(\prod_{i\in[d-1]} |P_i|^{x_i} \right) \binom{d}{x_1,\ldots,x_{d-1}} \\
&= C \cdot \frac{n^d}{d!}. \qquad\qquad\qquad\qquad\qquad\qquad\qquad\square
\end{aligned}
$$

4 Density-aware bounds

This section studies biclique partitions of graphs in terms of their edge density $\gamma := |E(G)|/\binom{|V|}{2}$. The number of graphs on n vertices with density γ is naturally $\binom{\binom{n}{2}}{\gamma\binom{n}{2}}$. We can understand this quantity through the following consequence of Stirling's approximation.

Remark 23. Let $\gamma_n \in (0,1)$ be a sequence such that $n\gamma_n \to \infty$ and $n(1-\gamma_n) \to \infty$. Then,

$$
\binom{n}{n\gamma_n} \sim \frac{2^{h_2(\gamma_n)n}}{\sqrt{2\pi\gamma_n(1-\gamma_n)n}},
$$

where $h_2(x) := -x\lg(x) - (1-x)\lg(1-x)$ is the binary entropy function.

A useful consequence of Remark 23 is that $\lg\binom{n}{\gamma n} \sim nh_2(\gamma)$. Thus, the information theoretic lower bound, as in Theorem 12, yields

$$
\mathsf{BP}(n,\gamma) \geqslant \left(\frac{1}{2} + o(1) \right) \cdot h_2(\gamma)\frac{n^2}{\lg n}.
$$

Our starting point for proving upper bounds in this setting is a nice result of Nechiporuk [Nec69].

16

Theorem 6 (Nechiporuk [Nec69]). *Let $G = (X \sqcup Y, E)$ be a bipartite graph with $|Y| \geqslant |X|$, and let $\gamma \in (0, 1)$ be such that $|X| = \omega(\frac{\lg |Y|}{h_2(\gamma)})$, $\max\{\gamma^{-1}, (1 - \gamma)^{-1}\} = |Y|^{o(1)}$, and $|E| = \gamma|X||Y|$. Then, G admits a biclique partition of weight $(1 + o(1)) \cdot h_2(\gamma) \frac{|X||Y|}{\lg |Y|}$, and this is asymptotically best possible.*

Now, similarly to the proof of Theorem 11 in the previous section, we will extend Nechiporuk's result to general graphs with an optimal constant by using tournaments.

For the proof, we will use the following combinatorial inequality.

Remark 24. For any pair of sequences of integers $a_1, \ldots, a_k$ and $b_1, \ldots, b_k$ with $a_i \geqslant b_i$, we have

$$\prod_{i=1}^{k} \binom{a_i}{b_i} \leqslant \binom{a_1 + \cdots + a_k}{b_1 + \cdots + b_k}.$$

To see why Remark 24 is true, consider that for choosing $B = b_1 + \cdots + b_k$ out of $A = a_1 + \cdots + a_k$ objects, one can always limit oneself to choose b_1 objects among the first a_1, b_2 among the second a_2 objects, and so on. The number of ways of choosing with this additional restriction, on the left side of Remark 24, cannot exceed the total number of ways $\binom{A}{B}$, on the right side.

Theorem 7. *Let $\gamma \in (0, 1)$ be such that $\max\{\gamma^{-1}, (1 - \gamma)^{-1}\} = n^{o(1)}$. Then, $BP(n, \gamma) \sim \frac{h_2(\gamma)}{2} \cdot \frac{n^2}{\lg n}$. Furthermore, given a graph represented as an adjacency list, a biclique partition realizing the upper bound can be constructed deterministically in time $O(m)$.*

Before proceeding with the proof, let us roughly sketch its main high-level idea dating back to [Nec69], and how Remark 24 comes into play. While in the proof of Theorem 11 we just split $V(G)$ into parts of equal size, the proof of Theorem 7 requires two levels of splitting: first we split $V(G)$ into parts $P_1, P_2, \ldots$ of a fixed equal size, this time depending on the density γ, and set up a tournament[3] between the parts just as in Theorem 11. But then, given a vertex v, we can view its adjacencies with respect to P_i as a binary string, which we further split into more granular *slices*, which do not have a fixed size; the slices are constructed so that they have roughly the same "entropy", instead of roughly the same size. The *entropy* of a string of length ℓ with Hamming weight w is $\lg \binom{\ell}{w}$. This way, after slicing the binary string b_v corresponding to v's adjacency row, and constructing bicliques that will take place within a slice, the total weight incurred will be at most

$$\sum_{v \in V} \sum_{\text{slice index } i} \lg \binom{\ell_i}{w_i} = \lg \left(\prod_{v \in V} \prod_{\text{slice index } i} \binom{\ell_i}{w_i}\right) \overset{\text{Rem. 24}}{\leqslant} (1 + o(1)) \lg \binom{n^2/2}{|E|} \overset{\text{Rem. 23}}{\sim} h_2(\gamma) \frac{n^2}{2},$$

which explains the shape of the obtained bound. Let us now proceed with the proof.

Proof of Theorem 7. The lower bound is via information theory as in the proof of Theorem 12, and it remains to prove the upper bound. Let G be an n-vertex graph with edge density γ, where $\max\{\gamma^{-1}, (1 - \gamma)^{-1}\} = n^{o(1)}$. Let $r := \left\lfloor \frac{\lg^2 n}{h_2(\gamma)} \right\rfloor$ and partition $V(G)$ into parts $P_1, \ldots, P_{\lceil n/r \rceil}$ of size at most r each. Given a vertex $v \in V(G)$, let $g(v)$ be the index of the part P_i containing v. Orient $[[n/r]]$ according to a tournament, and write $R(i, j)$ if there is a directed edge from i to j in this tournament.

---

[3]Note that it is unnecessary for the tournament to be almost regular for this application, since we only seek to bound the weight of the constructed biclique partition rather than the load on each vertex.

Label the vertices of G so that $V(G)$ is identified with the set $\{(i,j) : i \in [[n/r]], j \in [|P_i|]\}$. We write $(i,[a,b])$ as an abbreviation for $\{(i,j) : j \in [a,b]\}$. Given $v \in V(G)$, let $N_{i,a,b}(v) := N(v) \cap P_i \cap (i,[a,b])$. We define a strictly increasing sequence of indices $r(v,i,0), r(v,i,1), \ldots, r(v,i,\ell) \in [0,|P_i|]$ as follows. Let $r(v,i,0) = 0$. Then, if $r(v,i,j-1)$ has been defined and is not equal to $|P_i|$, let

$$r(v,i,j) = \max\left\{ x \in [r(v,i,j-1)+1, |P_i|] : \binom{x - r(v,i,j-1)}{|N_{i,r(v,i,j-1)+1,x}(v)|} < \frac{n}{r^4} \right\}. \tag{4}$$

With these indices defined, let

$$\mathsf{slices}(v,i) := \{(r(v,i,j-1)+1, r(v,i,j)) : j \in [\ell]\}.$$

Given $1 \leqslant a \leqslant b \leqslant |P_i|$ and nonempty $S \subseteq (i,[a,b])$, let

$$A_i(a,b,S) = \{v \in V(G) : N_{i,a,b}(v) = S \text{ and } (a,b) \in \mathsf{slices}(v,i) \text{ and } R(g(v),i)\},$$

and let

$$\mathcal{S}_{i,a,b} = \{\varnothing \neq S \subseteq (i,[a,b]) : A_i(a,b,S) \neq \varnothing\}.$$

Note that $(S, A_i(a,b,S))$ is a biclique for all $i \in [[n/r]]$, $1 \leqslant a \leqslant b \leqslant |P_i|$, and $S \in \mathcal{S}_{i,a,b}$. For ease of notation, let $\mathcal{I} := \{(i,a,b,S) : i \in [[n/r]], 1 \leqslant a \leqslant b \leqslant |P_i|, S \in \mathcal{S}_{i,a,b}\}$.

We define the following set of bicliques:

$$\mathcal{C} := \underbrace{\{(S, A_i(a,b,S)) : (i,a,b,S) \in \mathcal{I}\}}_{\text{across parts}} \cup \underbrace{\{(\{u\},\{v\}) : \{u,v\} \in E(G), g(u) = g(v)\}}_{\text{within parts}}$$

First, we claim that $\mathcal{C}$ is a biclique partition. Let $\{u,v\} \in E(G)$. If $g(u) = g(v)$, then $\{u,v\}$ is covered uniquely by the biclique $(\{u\},\{v\}) \in \mathcal{C}$. Otherwise, $g(u) \neq g(v)$, and we assume without loss of generality that $R(g(u), g(v))$. According to our labeling of $V(G)$, we have $v = (g(v), j)$ for some $j \in [|P_{g(v)}|]$. There is a unique $(a,b) \in \mathsf{slices}(u, g(v))$ with $j \in [a,b]$. Then, $\{u,v\}$ is covered uniquely by the biclique $(N_{g(v),a,b}(u), A_{g(v)}(a,b, N_{g(v),a,b}(u))) \in \mathcal{C}$.

Now, we bound the total weight of $\mathcal{C}$. For the following computations, note that our assumptions imply that $1/h_2(\gamma) = o(\gamma^{-1})$.

First, there are the bicliques of the form $(\{u\},\{v\})$ for $g(u) = g(v)$, whose total weight is at most $O([n/r] \cdot r^2) = O(n \lg^2(n)/h_2(\gamma)) = o(h_2(\gamma)\frac{n^2}{\lg n})$. Second, we have $\sum_{(i,a,b,S) \in \mathcal{I}} |S| \leqslant \sum_{(i,a,b,S) \in \mathcal{I}} r$. In this summation, we have $[n/r]$ choices for i, at most r^2 choices for a and b, at most r choices for $|S|$, and at most n/r^4 elements of $\mathcal{S}_{i,a,b}$ of the chosen size $|S|$. Hence,

$$\sum_{(i,a,b,S) \in \mathcal{I}} r \leqslant \frac{n^2}{r} = o\left(h_2(\gamma)\frac{n^2}{\lg n}\right).$$

It remains to show that

$$\sum_{(i,a,b,S) \in \mathcal{I}} |A_i(a,b,S)| \leqslant \left(\frac{1}{2} + o(1)\right) \cdot h_2(\gamma)\frac{n^2}{\lg n}.$$

We have

$$\sum_{(i,a,b,S) \in \mathcal{I}} |A_i(a,b,S)| = \sum_{(i,a,b,S) \in \mathcal{I}} \sum_{v \in V(G)} \mathbb{1}_{[v \in |A_i(a,b,S)|]}$$

$$= \sum_{v \in V(G)} \sum_{(i,a,b,S) \in \mathcal{I}} \mathbb{1}_{[v \in |A_i(a,b,S)|]}$$

$$= \sum_{v \in V(G)} |\{(a,b) \in \mathsf{slices}(v,i) : R(g(v),i)\}|.$$

Given $v \in V(G)$, $i \in [[n/r]]$, and $(a,b) \in \text{slices}(v,i)$ with $b < |P_i|$, we first observe that $0 < |N_{i,a,b}(v)| < b - (a-1)$; otherwise, we would have

$$\frac{n}{r^4} \leqslant \binom{(b+1) - (a-1)}{|N_{i,a,b+1}(v)|} = \binom{(b+1) - (a-1)}{1} \leqslant r,$$

a contradiction. Then, for any $(a,b) \in \text{slices}(v,i)$ with $b < |P_i|$, we have

$$\frac{n}{r^4} \leqslant \max\left\{ \binom{(b+1) - (a-1)}{|N_{i,a,b}(v)|}, \binom{(b+1) - (a-1)}{|N_{i,a,b}(v)| + 1} \right\}$$

$$\leqslant \binom{b - (a-1)}{|N_{i,a,b}(v)|} \cdot \frac{b - (a-1)}{\min\{|N_{i,a,b}(v)|, b - (a-1) - |N_{i,a,b}(v)|\}} \leqslant \binom{b - (a-1)}{|N_{i,a,b}(v)|} \cdot r,$$

from where

$$1 \leqslant \frac{\lg \binom{b-(a-1)}{|N_{i,a,b}(v)|}}{\lg(n) - \lg(r^5)}.$$

Thus,

$$\sum_{v \in V(G)} |\{(a,b) \in \text{slices}(v,i) : b < |P_i|, R(g(v),i)\}| \leqslant \sum_{\substack{v \in V(G) \\ i \in [[n/r]] \\ R(g(v),i)}} \sum_{\substack{(a,b) \in \text{slices}(v,i) \\ b < |P_i|}} \frac{\lg \binom{b-(a-1)}{|N_{i,a,b}(v)|}}{\lg(n) - \lg(r^5)}$$

$$= \frac{\lg\left(\prod_{\substack{v \in V(G) \\ i \in [[n/r]] \\ R(g(v),i)}} \prod_{\substack{(a,b) \in \text{slices}(v,i) \\ b < |P_i|}} \binom{b-(a-1)}{|N_{i,a,b}(v)|} \right)}{\lg(n) - \lg(r^5)}$$

$$\leqslant \frac{\lg \binom{\binom{n}{2}}{(\gamma \pm o(1))\binom{n}{2}}}{\lg(n) - \lg(r^5)} \qquad \text{(Remark 24)}$$

$$\leqslant \frac{h_2(\gamma \pm o(1)) \cdot \binom{n}{2}}{\lg(n) - \lg(r^5)} = \left(\frac{1}{2} + o(1) \right) \cdot h_2(\gamma) \frac{n^2}{\lg n}.$$

Therefore,

$$\sum_{v \in V(G)} |\{(a,b) \in \text{slices}(v,i) : R(g(v),i)\}| \leqslant \left(\frac{1}{2} + o(1) \right) \cdot h_2(\gamma) \frac{n^2}{\lg n} + \sum_{v \in V(G)} \lceil n/r \rceil$$

$$= \left(\frac{1}{2} + o(1) \right) \cdot h_2(\gamma) \frac{n^2}{\lg n},$$

as desired.

Finally, we show that the above construction can be carried out deterministically in time $O(m)$, assuming G is represented as an adjacency list. First, for each $y \in [r]$, we compute the maximum $x \in \mathbb{N}$ such that $\binom{x}{y} < \frac{n}{r^4}$ and store the result in a lookup table. After partitioning $V(G)$ into parts $P_1, \ldots, P_{\lceil n/r \rceil}$, we compute the list $N(v) \cap P_i$ for each $v \in V(G)$ and $i \in [n/r]$; this is so we can directly access $N(v) \cap P_i$ later without having to iterate through all of $N(v)$ each time. Now, for each $i \in [[n/r]]$, we maintain an array of linked lists A_i indexed by a, b, and S satisfying $1 \leqslant a \leqslant b \leqslant |P_i|$

and $\varnothing \neq S \subseteq (i, [a, b])$ such that $\binom{b-(a-1)}{|S|} < \frac{n}{r^4}$. Observe that $|A_i| < |P_i|^2 \cdot r \cdot \frac{n}{r^4} \leqslant \frac{n}{r}$, where the r term comes from the choices for $|S|$. Thus, adding over i, the total time allocating these arrays is $O(\frac{n^2}{r^2}) = O(\frac{n^2 h_2^2(\gamma)}{\lg^4 n})$, which is $O(m)$ since for γ bounded away from 0 we have $m = \Omega(n^2)$, and for $\gamma \to 0$ we have $h_2(\gamma) \sim \gamma \lg(1/\gamma)$, from where

$$O\left(\frac{n^2 h_2^2(\gamma)}{\lg^4 n}\right) = O\left(\frac{n^2 \cdot \gamma^2 \lg^2(1/\gamma)}{\lg^4 n}\right) = o\left(\frac{n^2 \gamma^2}{\lg^2 n}\right) = o(m).$$

Then, for each $v \in V(G)$ such that $R(g(v), i)$, we compute slices(v, i) according to (4) by iterating through $N(v) \cap P_i$ and using our lookup table constructed above; this takes $O(N(v) \cap P_i)$ time. For each $(a, b) \in$ slices(v, i), add v to $A_i(a, b, N(v) \cap (i, [a, b]))$. For each a, b, and S satisfying the above conditions, if $A_i(a, b, S) \neq \varnothing$, then output $(S, A_i(a, b, S))$; also, for each $\{u, v\} \in E(G[P_i])$, output $(\{u\}, \{v\})$. The complexity of the algorithm is thus

$$\sum_{\substack{v \in V(G) \\ i \in [[n/r]]}} O(N(v) \cap P_i) = O(m). \qquad \square$$

4.1 Density-aware Erdős–Pyber

Just as the Chung–Erdős–Spencer theorem can be strengthened to the Erdős–Pyber theorem, it is natural to ask if Theorem 7 admits a similar strengthening. That is, given a graph G of density γ, do we have $\mathsf{LBP}(G) \leqslant (h_2(\gamma)/2 + o(1))\frac{n}{\lg n}$? The following example shows that this is not possible in general.

Example 25. By Theorem 12, there is a graph H on $\lfloor n/8 \rfloor$ vertices such that

$$\mathsf{LBP}(H) \geqslant \frac{\mathsf{BC}(H)}{\lfloor n/8 \rfloor} \geqslant \left(\frac{1}{16} - o(1)\right)\frac{n}{\lg n}.$$

Let $G = H \sqcup \overline{K_{\lceil 7n/8 \rceil}}$, where $\overline{K_{\lceil 7n/8 \rceil}}$ is the empty graph on $\lceil 7n/8 \rceil$ vertices. Then, G is an n-vertex graph with edge density $\gamma \leqslant 1/64$ and $\mathsf{LBP}(G) = \mathsf{LBP}(H)$. Numerically, we have $h_2(1/64) < 1/8$, so

$$\mathsf{LBP}(G) > \left(\frac{h_2(\gamma)}{2} + o(1)\right)\frac{n}{\lg n}.$$

On the other hand, we can prove a density-aware version of the Erdős–Pyber theorem given a stronger assumption. Namely, we need bounds on the degree of each vertex rather than the total number of edges. Given $\dot\gamma \in [0, 0.5]$, let $\mathsf{LBP}(n, \dot\gamma) := \max\{\mathsf{LBP}(G) : |V(G)| = n$ and $\min\{d(v)/(n-1), 1 - d(v)/(n-1)\} \leqslant \dot\gamma$ for all $v \in V(G)\}$. Note that $\mathsf{LBP}(n, 0.5) = \mathsf{LBP}(n)$.

Theorem 26. *If $\dot\gamma \in [0, 0.5]$ and $\dot\gamma^{-1} = n^{o(1)}$, then*

$$LBP(n, \dot\gamma) \leqslant \left(\frac{1}{2} + o(1)\right) \cdot h_2(\dot\gamma)\frac{n}{\lg n}.$$

Furthermore, given a graph satisfying the degree constraints represented as an adjacency list, a biclique partition realizing the upper bound can be constructed in time $O(m)$ by a randomized algorithm with high probability.

20

Before proving the theorem, we need a lemma. For the following, let $d^+(v)$ be the outdegree of a vertex v in a directed graph.

Lemma 27. *Let G be an n-vertex graph such that $\min\{d(v)/(n-1), 1 - d(v)/(n-1)\} \leqslant \dot{\gamma} \leqslant 1/2$ for all $v \in V(G)$, where $\dot{\gamma}^{-1} = n^{o(1)}$. Let $r := \left\lfloor \frac{\lg^2 n}{h_2(\dot{\gamma})} \right\rfloor$ and suppose that $V(G)$ is partitioned into parts $P_1, \ldots, P_{\lceil n/r \rceil}$ of size at most r each. Given a vertex $v \in V(G)$, let $g(v)$ be the index of the part P_i containing v. Given a tournament R on $[[n/r]]$, write $d_R(v) := |\{w \in N_G(v) : R(g(v), g(w))\}|$. Then, there is a tournament R on $[[n/r]]$ such that (i) for every $i \in [[n/r]]$, we have $d^+(i) \leqslant (\frac{1}{2} + o(1))\lceil n/r \rceil$ and (ii) for every $v \in V(G)$, we have $\min\{d_R(v)/(n-1), 1 - d_R(v)/(n-1)\} \leqslant (\frac{1}{2} + o(1))\dot{\gamma}$.*

Proof. We claim that a tournament R in which the orientation of each edge is chosen uniformly and independently at random works with high probability. Note that our assumptions imply that $1/h_2(\dot{\gamma}) = o(\dot{\gamma}^{-1})$; in particular, $r = n^{o(1)}$.

First, we prove claim (i). Fix an element $i \in [[n/r]]$. Without loss of generality, take $i = 1$. Then, we have $d^+(i) = \sum_{j=2}^{\lceil n/r \rceil} \varepsilon_j$, where each $\varepsilon_j \in \{0, 1\}$ is a Bernoulli random variable. We have $\mathbb{E}[d^+(i)] \leqslant \frac{1}{2}\lceil n/r \rceil$. Let $\delta > 0$. Then, by Hoeffding's inequality,

$$\mathbb{P}\left[d^+(i) \geqslant \left(\frac{1}{2} + \delta\right)\lceil n/r \rceil\right] \leqslant \mathbb{P}\left[d^+(i) \geqslant \mathbb{E}[d^+(i)] + \delta\lceil n/r \rceil\right]$$

$$\leqslant \exp\left(\frac{-2(\delta\lceil n/r \rceil)^2}{\lceil n/r \rceil - 1}\right)$$

$$\leqslant \exp(-\delta^2 \omega(n^{1-\varepsilon}))$$

for all fixed $\varepsilon > 0$. Hence, by the union bound,

$$\mathbb{P}\left[d^+(i) \geqslant \left(\frac{1}{2} + \delta\right)\lceil n/r \rceil \text{ for some } i \in \lceil n/r \rceil\right] \leqslant \exp(-\delta^2 \omega(n^{1-\varepsilon}))$$

for all fixed $\varepsilon > 0$. Therefore, $d^+(i) \leqslant (\frac{1}{2} + o(1))\lceil n/r \rceil$ for all $i \in [[n/r]]$ with high probability.

Next, we prove claim (ii). Fix a vertex $v \in V(G)$. Without loss of generality, $g(v) = 1$ and $d(v)/(n-1) \leqslant \dot{\gamma}$ (the case where $1 - d(v)/(n-1) \leqslant \dot{\gamma}$ is symmetric). Then, we have $d_R(v) = \sum_{i=2}^{\lceil n/r \rceil} \varepsilon_i \cdot |N(v) \cap P_i|$, where each $\varepsilon_i \in \{0, 1\}$ is a Bernoulli random variable. Thus,

$$\mathbb{E}[d_R(v)] \leqslant \frac{1}{2}d(v) \leqslant \frac{1}{2}\dot{\gamma}(n-1).$$

Let $\delta > 0$. Then, by Hoeffding's inequality,

$$\mathbb{P}\left[d_R(v)/(n-1) \geqslant \left(\frac{1}{2} + \delta\right)\dot{\gamma}\right] \leqslant \mathbb{P}\left[d_R(v) \geqslant \mathbb{E}[d_R(v)] + \delta\dot{\gamma}(n-1)\right]$$

$$< \exp\left(\frac{-2(\delta\dot{\gamma}(n-1))^2}{(\lceil n/r \rceil - 1)r^2}\right)$$

$$\leqslant \exp(-\delta^2 \omega(n^{1-\varepsilon}))$$

for all fixed $\varepsilon > 0$. Hence, by the union bound,

$$\mathbb{P}\left[\min\{d_R(v)/(n-1), 1 - d_R(v)/(n-1)\} \geqslant \left(\frac{1}{2} + \delta\right)\dot{\gamma} \text{ for some } v \in V(G)\right] \leqslant \exp(-\delta^2 \omega(n^{1-\varepsilon}))$$

for all fixed $\varepsilon > 0$. Therefore, $\min\{d_R(v)/(n-1), 1 - d_R(v)/(n-1)\} \leqslant (\frac{1}{2} + o(1))\dot{\gamma}$ for all $v \in V(G)$ with high probability. $\qquad\square$

Proof of Theorem 26. Let G be an n-vertex graph such that $\min\{d(v)/(n-1), 1 - d(v)/(n-1)\} \leqslant \dot{\gamma} \leqslant 1/2$ for all $v \in V(G)$, where $\dot{\gamma}^{-1} = n^{o(1)}$. The construction of the biclique partition $\mathcal{C}$ is nearly identical to the one given in the proof of Theorem 7, the only difference being that rather than using an arbitrary tournament, we apply Lemma 27 to G to get a tournament R on $[[n/r]]$ such that $d^+(i) \leqslant (\frac{1}{2} + o(1))[n/r]$ for all $i \in [[n/r]]$ and $\min\{d_R(v)/(n-1), 1 - d_R(v)/(n-1)\} \leqslant (\frac{1}{2} + o(1))\dot{\gamma}$ for all $v \in V(G)$. In particular, the construction still takes $O(m)$ time, although it is now randomized.

Fix a vertex $v \in V(G)$, and we count how many bicliques from $\mathcal{C}$ contain v. For the following computations, note that our assumptions imply that $1/h_2(\dot{\gamma}) = o(\dot{\gamma}^{-1})$.

First, there are the bicliques of the form $(\{u\}, \{v\})$ for $g(u) = g(v)$, of which there are at most $O(|P_{g(v)}|) = O(\lg^2(n)/h_2(\dot{\gamma})) = o(h_2(\dot{\gamma})\frac{n}{\lg n})$. Second, there are the bicliques of the form $(S, A_i(a, b, S))$ with $v \in S$. We upper bound this quantity by counting how many bicliques of the form $(S, A_i(a, b, S))$ with $\varnothing \neq S \subseteq P_{g(v)}$ there are. There are at most r^2 choices for a and b, at most r choices for $|S|$, and at most n/r^4 elements of $\mathcal{S}_{i,a,b}$ of the chosen size $|S|$. Hence, the number of bicliques of the form $(S, A_i(a, b, S))$ with $v \in S$ is at most $O(n/r) = o(h_2(\dot{\gamma})\frac{n}{\lg n})$. Third, there are the bicliques of the form $(S, A_i(a, b, S))$ with $v \in A_i(a, b, S)$, which we now proceed to analyze.

For any $(a, b) \in \mathsf{slices}(v, i)$ with $b < |P_i|$, we first observe that $0 < |N_{i,a,b}(v)| < b - (a - 1)$; otherwise, we would have

$$\frac{n}{r^4} \leqslant \binom{(b+1) - (a-1)}{|N_{i,a,b+1}(v)|} = \binom{(b+1) - (a-1)}{1} \leqslant r,$$

a contradiction. Then, for any $(a, b) \in \mathsf{slices}(v, i)$ with $b < |P_i|$, we have

$$\frac{n}{r^4} \leqslant \max\left\{ \binom{(b+1) - (a-1)}{|N_{i,a,b}(v)|}, \binom{(b+1) - (a-1)}{|N_{i,a,b}(v)| + 1} \right\}$$
$$\leqslant \binom{b - (a-1)}{|N_{i,a,b}(v)|} \cdot \frac{b - (a-1)}{\min\{|N_{i,a,b}(v)|, b - (a-1) - |N_{i,a,b}(v)|\}} \leqslant \binom{b - (a-1)}{|N_{i,a,b}(v)|} \cdot r,$$

from where

$$1 \leqslant \frac{\lg\binom{b-(a-1)}{|N_{i,a,b}(v)|}}{\lg(n) - \lg(r^5)}.$$

Thus,

$$|\{(a, b) \in \mathsf{slices}(v, i) : b < |P_i|, R(g(v), i)\}| \leqslant \sum_{\substack{i \in [[n/r]] \\ R(g(v),i)}} \sum_{\substack{(a,b) \in \mathsf{slices}(v,i) \\ b < |P_i|}} \frac{\lg\binom{b-(a-1)}{|N_{i,a,b}(v)|}}{\lg(n) - \lg(r^5)}$$

$$= \frac{\lg\left(\prod_{\substack{i \in [[n/r]] \\ R(g(v),i)}} \prod_{\substack{(a,b) \in \mathsf{slices}(v,i) \\ b < |P_i|}} \binom{b-(a-1)}{|N_{i,a,b}(v)|} \right)}{\lg(n) - \lg(r^5)}$$

$$\leqslant \frac{\lg\binom{(1/2+o(1))n}{d_R(v)}}{\lg(n) - \lg(r^5)} \qquad \text{(Remark 24)}$$

$$\leqslant \frac{\left(\frac{1}{2} + o(1)\right) \cdot h_2(\dot{\gamma}) \cdot n}{\lg(n) - \lg(r^5)} = \left(\frac{1}{2} + o(1)\right) \cdot h_2(\dot{\gamma})\frac{n}{\lg n}.$$

22

Therefore, the number of bicliques of the form $(S, A_i(a, b, S))$ with $v \in A_i(a, b, S)$ is

$$|\{(a, b) \in \text{slices}(v, i) : R(g(v), i)\}| \leqslant \left(\frac{1}{2} + o(1)\right) \cdot h_2(\dot{\gamma}) \frac{n}{\lg n} + \lceil n/r \rceil$$

$$= \left(\frac{1}{2} + o(1)\right) \cdot h_2(\dot{\gamma}) \frac{n}{\lg n},$$

as desired. $\square$

5 Biclique representation of graphs

An interesting interpretation of Theorem 7 is that representing graphs by biclique partitions is information theoretically optimal for any density γ such that $\max\{\gamma^{-1}, (1 - \gamma)^{-1}\} = n^{o(1)}$. In general, when representing objects from a set $\mathcal{U}$, a representation is said to be *succinct* if it uses $\lg(\mathcal{U})(1 + o(1))$ bits, and *compact* if it uses $O(\lg(\mathcal{U}))$ bits [Nav16]. We consider the following two biclique-based representations for an undirected graph G:

1. **SB** (Succinct Biclique Representation). Given a biclique partition $\mathcal{B} = \{(L_1, R_1), \ldots, (L_k, R_k)\}$ of G, we store a list $L_{\mathcal{B}}$ of bicliques, where each biclique is stored as a structure with two sorted lists of vertices, corresponding to its sides.

2. **CB** (Compact Biclique Representation). On top of the SB representation, we store as well for each vertex v, a list of pointers to the bicliques (L_i, R_i) it belongs to, and also one extra bit per biclique representing whether v belongs to L_i or to R_i.

Listing 1: Example of a concrete data-layout for biclique representations.

```
/* ----- SB: Succinct Biclique Representation ----- */
typedef struct {
    int *L;      /* left  side (sorted vertex ids)  */
    int  L_len;
    int *R;      /* right side (sorted vertex ids)  */
    int  R_len;
} Biclique;

typedef struct {
    int  n;      /* number of vertices [0..n-1] */
    int  k;      /* number of bicliques         */
    Biclique *B;    /* array of k bicliques        */
} SB;

/* ----- CB: Compact Biclique Representation  ----- */
typedef struct {
    int  biclique;  /* index into sb.B (0..sb.k-1) */
    bool on_left;   /* bit for the side in which the vertex is */
} Incidence;

typedef struct {
    Incidence *a;   /* incidences for a vertex */
    int   len;
} IncList;

typedef struct {
```

```
    SB        sb;     /* the underlying SB representation */
    IncList *inc;     /* array of sb.n per-vertex lists  */
} CB;
```

Naturally, for these representations we will compute a small biclique partition leveraging Theorem 7, or sometimes Theorem 11. The total size of the SB representation, using a biclique partition $\mathcal{B}$ is $w(\mathcal{B}) \cdot \lg n$ (each vertex id uses $\lg n$ bits), and thus by Theorem 7, it allows to represent any n-vertex graph of density γ with only $(1/2 + o(1)) \cdot h_2(\gamma)n^2$ bits, which indeed makes it a succinct representation. On the other hand, the CB representation uses three times as many bits: since there are no more than $O(n^2)$ bicliques in $\mathcal{B}$, each pointer can be stored in $2 \lg n$ bits, and thus, if $\ell_\mathcal{B}(v)$ is the number of bicliques a vertex v belongs to, then the additional space used by the CB representation is

$$\sum_{v \in V(G)} \ell_\mathcal{B}(v) \cdot (2\lg n + 1) \sim 2\lg n \sum_{v \in V(G)} \sum_{(L_i, R_i) \in \mathcal{B}} \mathbb{1}_{[v \in (L_i \cup R_i)]} = 2\lg n \sum_{(L_i, R_i) \in \mathcal{B}} |L_i| + |R_i| = 2\lg n \cdot w(\mathcal{B}).$$

While alternative succinct and compact representations for graphs are known [Nav16, FM13], we will show a few examples for which these biclique representations can result in algorithmic improvements. Furthermore, by representing bicliques with a succinct data structure for integer sequences, such as the one given by Golynski, Munro, and Rao [GMR06], it is possible to transform the CB representation into a succinct one at the cost of accessing the i-th biclique of a given vertex in time $O(\lg \lg n)$ instead of $O(1)$. The main idea, as in the work of Hernández and Navarro [HN13], is that succinct data structures for an integer sequence S still allow to efficiently identify the position of the i-th occurrence of an integer v in S, a query known as $\text{select}(v, i)$, and thus if S is the concatenation of the biclique parts, then identifying the i-th occurrence of v in S corresponds to identifying the i-th biclique in which v appears.

5.1 Independent set queries

Suppose that, given a graph $G = (V, E)$, we wish to answer a sequence of *independent set queries* of the form "Is $S \subseteq V$ an independent set?". This problem was studied first by Williams [Wil07], who showed that by preprocessing the graph in time $O(n^{2+\varepsilon})$, for any $\varepsilon > 0$, one can answer each query in time $O(n^2/(\varepsilon \lg n)^2)$. This was subsequently improved by a randomized preprocessing algorithm of Bansal and Williams [BW09], who also discussed interesting algorithmic implications, and more recently derandomized by Vassilevska Williams and Williams [WW18], resulting in the following theorem.

Theorem 28 ([WW18], [BW09, Thm. 2.3]). *There is a deterministic algorithm that preprocesses an n-vertex graph in time $O(n^{2+\varepsilon})$ for any $\varepsilon > 0$, so that any batch of $\lg n$ independent set queries can be answered deterministically in time $O(\frac{1}{\varepsilon}n^2(\lg \lg n)^2/(\lg n)^{5/4})$.*

It turns out that, by storing a graph in the SB representation, which can be computed in time $O(n^2)$ deterministically by Theorem 11, we can answer independent set queries in time $O(n^2/\lg n)$, which improves on the naive $\Omega(|S|^2) = \Omega(n^2)$ algorithm. Note that, if the number of independent set queries is very small, say $O(\lg \lg n)$, and we want to limit the preprocessing time to e.g., $O(n^2 \lg n)$, then using Theorem 28 as a black box yields a runtime of $O(n^2 \lg \lg n / \lg^{1/4} n)$ by taking $\varepsilon = \frac{\lg \lg n}{\lg n}$, whereas our algorithm runs in $O(n^2 \lg \lg n / \lg n)$, and with strictly less preprocessing.

Proposition 29. *Given an SB representation of an n-vertex graph based on a partition $\mathcal{B}$, an independent set query S can be answered in time $O(w(\mathcal{B}) + |S|)$.*

24

Proof. Let $\mathcal{B} = \{(L_1, R_1), \ldots, (L_k, R_k)\}$. Note that a subset $S \subseteq V$ is independent if and only if there is no biclique $(L_i, R_i) \in \mathcal{B}$ such that $|L_i \cap S| > 0$ and $|R_i \cap S| > 0$. Therefore, we can create a bitvector $B_S[1..n]$ such that $B_S[v] = 1$ if $v \in S$ and $B_S[v] = 0$ otherwise. This takes $O(|S|)$ time, since it suffices to iterate over the elements of S and mark their corresponding positions in B_S. Then for each i we check whether $|S \cap L_i| > 0$ in time $O(|L_i|)$ simply by iterating over the vertices of $v \in L_i$, and checking whether $B_S[v] = 1$ for at least some v. We proceed analogously for R_i. The total runtime is thus

$$O(|S|) + \sum_{i=1}^{k} O(|L_i|) + O(|R_i|) = O\left(|S| + \sum_{i=1}^{k} |L_i| + |R_i|\right) = O(w(\mathcal{B}) + |S|). \qquad \square$$

5.2 Cut queries

We now consider queries in which we are given two disjoint subsets of vertices, $S, T \subseteq V$, and we wish to answer the number $c(S, T)$ of edges with one endpoint in S and the other one in T. Applications of these queries to other algorithmic problems, in the particular case where $T = V \backslash S$, have been discussed by Apers et al. [AEG⁺22], and Lee, Santha, and Zhang [LSZ21]. The case of arbitrary disjoint subsets S, T is used, for instance, by Assadi et al. [ACK21].[4] Note that if $\mathbf{A}_G$ is the adjacency matrix of the graph, and $\mathbf{x}_S$ and $\mathbf{x}_T$ are vectors representing S and T respectively, then $c(S, T) = \mathbf{x}_S^\top \mathbf{A}_G \mathbf{x}_T$. Unfortunately, the algorithms for matrix-vector multiplication achieving the runtime of Theorem 28 work over the Boolean semi-ring, and thus they cannot be used as a black box for computing these products over $\mathbb{N}$. Our Proposition 29 can be trivially extended to cut queries as follows.

Proposition 30. *Given an SB representation of an n-vertex graph based on a partition $\mathcal{B}$, a cut query can be answered in time $O(w(\mathcal{B}) + |S| + |T|)$.*

Proof. Let $\mathcal{B} = \{(L_1, R_1), \ldots, (L_k, R_k)\}$, and observe that $c(S, T) = \sum_{i=1}^{k} |S \cap L_i| \cdot |T \cap R_i| + |S \cap R_i| \cdot |T \cap L_i|$. As in the proof of Proposition 29, these intersections can be computed in $O(w(\mathcal{B}) + |S| + |T|)$ by constructing corresponding bitvectors B_S and B_T. $\qquad \square$

5.3 Densest subgraph approximations

We now present a more interesting application of these biclique representations, which requires the CB representation.

The *densest subgraph problem* asks for a non-empty subset $S \subseteq V$ that maximizes the degree density $\delta(S) := |E(G[S])|/|S|$. Algorithms for this problem are widely used in many practical scenarios, as for instance identifying dense clusters in social media networks. For a survey on this problem and its applications, we refer the reader to Lanciano et al. [LMFB24].

Arguably, the most famous results regarding the densest subgraph problem were proved by Charikar [Cha00], who proved that an optimal solution could be computed by linear programming, and that a 2-approximation[5] could be computed in linear time (i.e., $\Theta(|V| + |E|)$), through an elegant greedy peeling algorithm. Charikar's algorithm is simply: start with $H_0 := G$, and then iteratively remove the vertex of smallest degree until the graph is empty (so $H_{i+1} :=$

---

[4]Note that the case $c(S, V \backslash S)$ is not easier since $2 \cdot c(S, T) = c(S, V \backslash S) + c(T, V \backslash T) - c(S \cup T, V \backslash (S \cup T))$.

[5]We use 2-approximation for consistency with the literature on the problem, but it is technically a $\frac{1}{2}$-approximation since it is a maximization problem.

$H_i \backslash \{\arg\min_{v \in V(H_i)} \deg(v)\}$), and finally return the densest of all the subgraphs H_i seen throughout the process.

In this subsection we show that $o(|V|^2)$ approximation algorithms are possible if the graph is stored in the CB representation, by modifying Charikar's algorithm.

Theorem 8. *Given a CB representation for an n-vertex graph G, where the biclique partition has weight $O(n^2 / \lg n)$, and any $\alpha > 1$, one can compute a 2α-approximation for the densest subgraph problem in time $O(n^2 / \lg \alpha)$.*

For instance, setting $\alpha = \lg \lg n$, we get a $2 \lg \lg n$ approximation in time $O(n^2 / \lg \lg \lg n) = o(n^2)$.

Proof of Theorem 8. The algorithm, whose pseudocode is presented in Algorithm 2, is parameterized by the desired $\alpha > 1$, and proceeds by rounds. Let H_i be the graph at the beginning of round i,

---

Algorithm 2 Densest Subgraph Approximation

Require: Graph $G = (V, E)$, assuming wlog that $V = \{1, \dots, n\}$. Approximation parameter $\alpha > 1$
1: maxDensity $\leftarrow 0$
2: densestSubgraph $\leftarrow \varnothing$
3: $t \leftarrow 1$
4: $H \leftarrow \text{copy}(G)$
5: removed $\leftarrow [\text{false}, \dots, \text{false}]$ ▷ *Array of length n, indexed from 1*
6: **while** $|V(H)| > 0$ **do**
7:      degreeSum $\leftarrow 0$
8:      toRemove $\leftarrow \varnothing$
9:      **for all** $v \in \{1, \dots, n\}$ **do** ▷ *In $O(n^2 / \lg n)$ time*
10:          **if** removed$[v]$ = false **then**
11:              $d \leftarrow H.\text{degree}(v)$
12:              degreeSum $\leftarrow$ degreeSum $+ d$
13:              **if** $d < t$ **then**
14:                  toRemove $\leftarrow$ toRemove $\cup \{v\}$
15:      currDensity $\leftarrow$ degreeSum$/(2|V(H)|)$
16:      **if** currDensity $>$ maxDensity **then**
17:          maxDensity $\leftarrow$ currDensity
18:          densestSubgraph $\leftarrow \text{copy}(H)$ ▷ *In $O(n^2 / \lg n)$ time*
19:      **for all** $v \in$ toRemove **do** ▷ *In $O(n^2 / \lg n)$ time*
20:          $H.\text{lazyRemove}(v)$ ▷ *v will no longer be counted for the degree of other vertices*
21:          removed$[v] \leftarrow$ true
22:      $t \leftarrow \alpha \cdot t$ ▷ *Increase threshold*

---

with $H_0 := G$. Then, at the end of round $i \geq 0$, all vertices whose degree in H_i is less than α^i are deleted, resulting in a subgraph H_{i+1}. Naturally, the process stops when all vertices get removed due to having degree less than α^k for some k, and then the algorithm returns the densest among all the subgraphs H_i seen throughout the rounds. To avoid introducing complicated data structures, instead of actually removing vertices from the graph, we will simply *mark them as removed* in a way that ensures that they will not be considered in further rounds. More concretely, we will implement an operation lazyRemove(v) such that a vertex v lazily removed will no longer count toward the degree of other vertices.

First, we analyze the runtime of Algorithm 2. Since each vertex has degree at most $n - 1$ in G, the algorithm stops after k rounds when $\alpha^k > n - 1$, and thus in no more than $\lg_\alpha(n)$ rounds. The

runtime of each round of Algorithm 2 is $O(n^2/\lg n)$ as long as we can query the degree of any number of vertices in $O(n^2/\lg n)$ as well as lazily deleting any number of vertices in $O(n^2/\lg n)$, and copying the graph in $O(n^2/\lg n)$. Thus, provided we can support such operations with the described runtime, the total runtime is $O(n^2/\lg n) \cdot \lg_\alpha(n) = O(n^2/\lg \alpha)$ as desired.

Claim 31. On a CB representation of weight w, we can query the degree of all vertices in time $O(w)$, and lazily delete an arbitrary subset of them in $O(w)$.

Proof of Claim 31. To compute the degree of a vertex v, it suffices to iterate over the list of bicliques it belongs to (i.e., `IncList` in Listing 1), and then for each such biclique $B = (L, R)$, if v is on its left side (i.e., `on_left`), then we add $|R|$ (i.e., `R_len`) to its degree, and otherwise $|L|$ (i.e., `L_len`). This takes time proportional to the number of bicliques containing v, or in other words, its load $\ell(v)$. Therefore the total runtime is $O(\sum_v \ell(v)) = O(w)$. For the lazy deletion of a vertex v, it suffices to iterate over the bicliques $B = (L, R)$ containing v, and if $v \in L$ we decrease `L_len` by 1, and otherwise `R_len` by 1. Since the degree computation of further vertices will be based on adding up `L_len` or `R_len`, this ensures future degree computations will be correct. The runtime is again $O(\sum_v \ell(v)) = O(w)$. Finally, since all indices and pointers use $\Theta(\lg n)$ bits, and thus constantly many RAM words, the cost of copying is proportional to the number of indices and pointers, which is $O(w)$. *(End of proof of Claim 31)* ∎

Now we prove correctness. First, we will need a standard idea from the densest-subgraph literature (see [LMFB24, Theorem 3]). We will use notation $\deg(v, S)$ for the degree of a vertex v in the subgraph $G[S]$.

Claim 32. Let $\delta^\star := \max\{\delta(S) : S \subseteq V(G)\}$, and $S^\star \subseteq V$ a subset for which $\delta(S^\star) = \delta^\star$. Then, for every $v \in S^\star$, we have $\deg(v, S^\star) \geqslant \delta^\star$.

Proof of Claim 32. Let v be any vertex in $S^\star$. By the optimality of $S^\star$ we have $\delta(S^\star) \geqslant \delta(S^\star \setminus \{v\})$, and thus $\frac{|E(G[S^\star])|}{|S^\star|} \geqslant \frac{|E(G[S^\star])| - \deg(v, S^\star)}{|S^\star| - 1}$, from where $\deg(v, S^\star) \geqslant |E(G[S^\star])| / |S^\star| = \delta^\star$. *(End of proof of Claim 32)* ∎

Now observe that at the end of round i, all vertices of degree less than α^i have been removed, and thus $\deg(v, V(H_{i+1})) \geqslant \alpha^i$ for every $v \in V(H_{i+1})$, and thus by the handshake lemma, $|E(H_{i+1})| \geqslant \frac{\alpha^i}{2} \cdot |V(H_{i+1})|$. Therefore, as long as $|V(H_{i+1})| > 0$, we have

$$\delta(V(H_{i+1})) = \frac{|E(H_{i+1})|}{|V(H_{i+1})|} \geqslant \frac{\alpha^i}{2}. \tag{5}$$

Let $i^\star$ be such that $1 \leqslant \alpha^{i^\star} \leqslant \delta^\star < \alpha^{i^\star+1}$, which exists since $i \mapsto \alpha^i$ is strictly increasing[6]. Then, note that we have $S^\star \subseteq V(H_{i^\star+1})$ since by Claim 32 all vertices in $S^\star$ have degree at least $\delta^\star$ within $S^\star$ and thus could not be deleted during round $i^\star$ since the threshold was $\alpha^{i^\star} \leqslant \delta^\star$. As clearly $|S^\star| > 0$, Equation (5) gives

$$\delta(V(H_{i^\star+1})) \geqslant \frac{\alpha^{i^\star}}{2} = \frac{\alpha^{i^\star+1}}{2\alpha} > \frac{\delta^\star}{2\alpha}.$$

As $H_{i^\star+1}$ is one of the subsets considered by the algorithm, we conclude that Algorithm 2 returns a 2α-approximation. Together with the runtime analysis above, this concludes the proof. □

---

[6]Here we assume without loss of generality that $\delta^\star \geqslant 1$, since any cycle has degree density 1, and for a tree G on n vertices the densest subgraph is G itself, with density $1 - 1/n$, which can be easily checked in linear time.
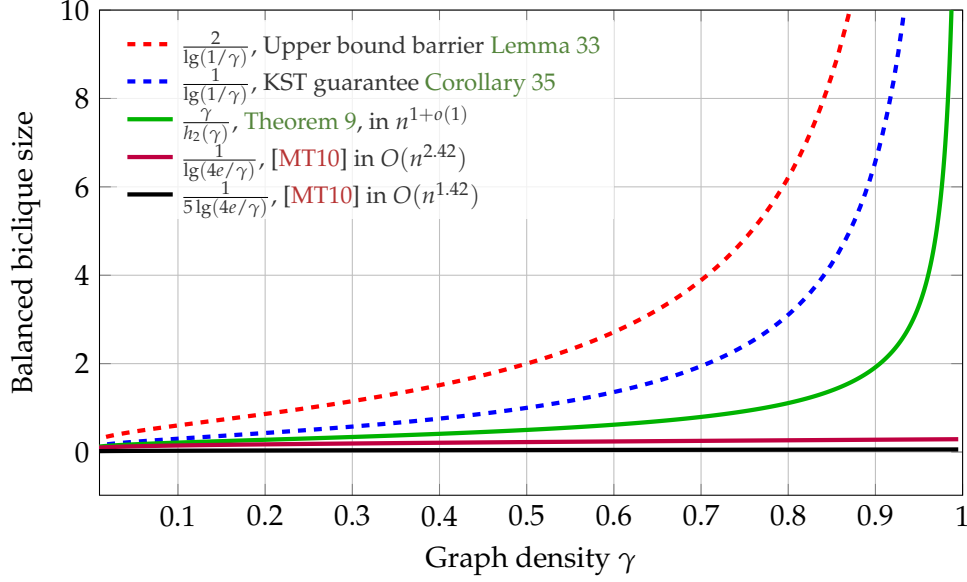
Figure 5: Comparison of the leading constant for $\Omega(\lg n)$ balanced bicliques, and algorithmic runtimes.

6 Finding large balanced bicliques

In this section, we study the algorithmic problem of finding a balanced biclique $K_{t,t}$, for t as large as possible, that is guaranteed by the edge density $\gamma := |E|/\binom{|V|}{2}$ of the input graph. In order to contextualize our results, we will consider known upper and lower bounds, as well as previous results of Mubayi and Turán [MT10]. These are summarized in Figure 5.

First, we shall show a limit to the largest biclique we can aim to find based only on edge density.

Lemma 33. *Let $\gamma \in (0,1)$. Then, with high probability, a random graph $G \sim G(n,\gamma)$ has edge density $\gamma \pm o(1)$ and yet does not contain a $K_{t,t}$ with $t \geqslant \frac{2}{\lg(1/\gamma)} \lg n$.*

Proof. The number of edges is $\gamma\binom{n}{2} \pm o(n^2)$ with high probability by a standard Chernoff bound. For the second part, we have $\mathbb{P}[G \text{ contains } K_{t,t}] < \binom{n}{t}\binom{n}{t}\gamma^{t^2} \leqslant \left(\frac{en}{t}\right)^{2t}\gamma^{t^2}$, and thus to show this probability goes to 0 it suffices to show that $\lg\left(\left(\frac{en}{t}\right)^{2t}\gamma^{t^2}\right) \to -\infty$, and equivalently,

$$2t \lg(n) + 2t \lg(e) - 2t \lg t - t^2 \lg(1/\gamma) \to -\infty.$$

But for $t \geqslant \frac{2}{\lg(1/\gamma)} \lg n$ we have $t^2 \lg(1/\gamma) \geqslant 2t \lg(n)$, so

$$2t \lg(n) + 2t \lg(e) - 2t \lg t - t^2 \lg(1/\gamma) \leqslant 2t \lg(e) - 2t \lg t \to -\infty. \qquad \square$$

In the opposite direction, we use upper bounds on the extremal problem $\mathrm{ex}(n, K_{t,t})$, the maximum number of edges in a $K_{t,t}$-free graph on n-vertices. While there have been improvements for many combinations of parameters since the original KST theorem (for a nice survey, see [FS13]), the following will be enough for our purposes:

Theorem 34 ([KST54], as stated by [FS13, Thm. 2.22]). *The maximum number of edges in a $K_{t,t}$-free graph G with n vertices, denoted $\mathrm{ex}(n, K_{t,t})$, satisfies*

$$\mathrm{ex}(n, K_{t,t}) \leqslant \frac{1}{2}(t-1)^{1/t}n^{2-1/t} + \frac{1}{2}(t-1)n.$$

28

Corollary 35. *Every graph on n vertices with edge density $\gamma < 1$ such that $\max\{\gamma^{-1}, (1-\gamma)^{-1}\} = n^{o(1)}$, contains a $K_{t,t}$ with $t \geqslant (1 - o(1))\frac{\lg n}{\lg(1/\gamma)}$.*

Proof. Let $t := \left\lfloor \frac{\lg n - \lg\lg n + \lg\lg(1/\gamma)}{\lg(1/\gamma)+1} \right\rfloor$, and note that as $t = (1-o(1))\frac{\lg n}{\lg(1/\gamma)}$ as desired. Then, observe that

$$t \leqslant \frac{\lg n - \lg\lg n + \lg\lg(1/\gamma)}{\lg(1/\gamma)+1} = \frac{\lg n - \lg(\frac{\lg n}{\lg(1/\gamma)})}{\lg(2/\gamma)} \leqslant \frac{\lg n - \lg t}{\lg(2/\gamma)},$$

from where $\lg(2/\gamma) \leqslant \frac{\lg n - \lg t}{t}$, and thus $\lg(\gamma/2) \geqslant \frac{\lg t}{t} - \frac{\lg n}{t} = \lg(t^{1/t}n^{-1/t})$. By exponentiating, we obtain $\gamma/2 \geqslant t^{1/t}n^{-1/t} \geqslant (t-1)^{1/t}n^{-1/t}$. Moreover, since $\gamma^{-1} = n^{o(1)}$, we have for large n that $\gamma/2 \geqslant t/n$. Adding this and the previous inequality, and multiplying by $\frac{n^2}{2}$, we get

$$\gamma\frac{n^2}{2} \geqslant \frac{1}{2}(t-1)^{1/t}n^{2-1/t} + \frac{tn}{2} \geqslant \frac{1}{2}(t-1)^{1/t}n^{2-1/t} + \frac{(t-1+\gamma)n}{2},$$

from where by Theorem 34, $E(G) = \gamma\binom{n}{2} \geqslant \mathrm{ex}(n, K_{t,t})$. $\qquad\square$

In terms of previous results, Mubayi and Turán [MT10] presented two algorithmic results: one that finds a balanced biclique of size $(1 - o(1))\frac{\lg n}{\lg(4e/\gamma)}$ in time $O(n^{2.42})$, and the second finds a balanced biclique 5 times smaller than the first one but in time $O(n^{1.42})$ assuming the degree of each vertex can be queried in $O(1)$, and $O(m)$ otherwise. We will present more efficient algorithms that also yield larger bicliques for almost all values of γ, but especially when γ is not constant and tends to 1, as illustrated in Figure 5.

An important difference between our algorithms and those of Mubayi and Turán is that their algorithms can never find a balanced biclique of size $\omega(\lg n)$ in polynomial time, even when $\gamma \to 1$, since their algorithm explores $\Omega(2^t)$ subsets for finding a $K_{t,t}$. In contrast, leveraging the density-aware biclique partitions we obtain from Theorem 7, we can obtain bicliques of size e.g., $\Omega(\lg^2 n / \lg\lg n)$ when $\gamma \geqslant 1 - \frac{1}{\lg n}$, as we will show next.

First, we notice that while Chung, Erdős, and Spencer leveraged the existence of large bicliques guaranteed by the KST theorem in order to find a biclique partition of small weight, the following observation shows that the implication can be reversed: finding any biclique partition of small weight implies finding a relatively large biclique.

Lemma 36 ([CY25, Observation 1]). *Let $\mathcal{B}$ be a biclique cover of a graph G. Then, $\mathcal{B}$ contains a biclique $K_{a,b}$ with $\min\{a,b\} \geqslant |E|/w(\mathcal{B})$.*

Proof. Let $\{(L_1, R_1), \ldots, (L_k, R_k)\}$ be a biclique cover for G of weight $w(\mathcal{B})$ and with $|L_i| \leqslant |R_i|$ for every i. Assume expecting a contradiction that G does not contain a $K_{t,t}$ for $t \geqslant |E|/w(\mathcal{B})$, and thus $|L_i| < |E|/w(\mathcal{B})$ for each i. Then,

$$|E| \leqslant \sum_{i=1}^{k}|L_i| \cdot |R_i| < \sum_{i=1}^{k}\frac{|E|}{w(\mathcal{B})}|R_i| \leqslant \sum_{i=1}^{k}\frac{|E|}{w(\mathcal{B})}(|L_i| + |R_i|) = |E|,$$

a contradiction. $\qquad\square$

Using the deterministic $O(m)$ algorithm from Theorem 7, we obtain a biclique partition $\mathcal{B}$ of weight at most $(1 + o(1))\frac{h_2(\gamma)n^2}{2\lg n}$, and then simply checking for the largest balanced biclique in $\mathcal{B}$,

29

we obtain a $K_{t,t}$ which according to Lemma 36 satisfies

$$t \geqslant \frac{\gamma \binom{n}{2}}{(1+o(1))\frac{h_2(\gamma)n^2}{2\lg n}} \geqslant (1-o(1))\frac{\gamma}{h_2(\gamma)}\lg n.$$

This is a stronger quantitative guarantee than we get with Mubayi and Turán's algorithm while still being efficient. In fact, building the entire biclique partition given by Theorem 7 is unnecessary if we only want to find a single $K_{t,t}$ of the desired size, so we can actually find such a biclique in time $n^{1+o(1)}$ assuming the right computational model. Without randomness or $O(1)$ access to degrees, it is easy to show a lower bound of $\Omega(n^2)$ when γ is fixed: if one queries $o(n^2)$ edges, it can be that none of them are present even if the graph has positive density. Therefore, we consider both randomized algorithms without $O(1)$ degree queries and deterministic algorithms with $O(1)$ degree queries. We assume an adjacency matrix representation for the input graph. We start with a couple lemmas.

Lemma 37. *Let $G = (V, E)$ be a graph on n vertices and edge density γ such that $\gamma^{-1} = n^{o(1)}$, and $r = O(\lg^k n)$ for some k be an integer. Then, if $D \subseteq V$ consists of the r vertices with highest degree, then the number of edges between D and $V \backslash D$ is at least $(1-o(1))r(n-r)\gamma$ with the $o(1)$ term being $O\left(\frac{r}{\gamma n}\right)$.*

Proof. Let us use notation $\overline{\deg}(S) := \frac{1}{|S|}\sum_{v\in S}\deg(v)$ for the average degree of a set of vertices. We trivially have $\overline{\deg}(D) \geqslant \overline{\deg}(V) = \frac{2|E|}{n} = \gamma(n-1)$. Therefore, denoting by $e(A, B)$ the number of edges between sets A and B, and $e(A)$ the number of edges with both endpoints in A, we have

$$e(D, V\backslash D) = \left(\sum_{v\in D}\deg(v)\right) - 2e(D) \geqslant r\,\overline{\deg}(V) - r^2 = r(n-1)\gamma - r^2.$$

Since

$$r(n-1)\gamma - r^2 \geqslant r(n-r)\gamma - r^2 = r(n-r)\gamma\left(1 - \frac{r}{\gamma(n-r)}\right),$$

and clearly $\frac{r}{\gamma(n-r)} = O(\frac{r}{\gamma n}) = o(1)$, this concludes the proof. $\qquad\square$

Lemma 38. *Let $G = (V, E)$ be a graph on n vertices with edge density γ such that $\gamma^{-1} = n^{o(1)}$, and $r = O(\lg^k n)$ for some k an integer. Then, there is a randomized algorithm, running in time $o(n)$ w.h.p. that finds a subset $D \subset V$ of size r such that with high probability there are $\gamma r(n-r)(1 - O(\lg^{-1/3} n))$ edges between D and $V \backslash D$.*

Proof. Let $\varepsilon := 1/\sqrt[3]{\lg n}$. The algorithm proceeds as follows:

1. Initialize $D := \varnothing$.

2. Sample a vertex $v \in V$ uniformly at random.

3. Estimate $\deg(v)$ by sampling a set $U \subset V\backslash\{v\}$ of size $|U| = \lceil\sqrt{n}\rceil$ uniformly at random, and taking the estimate $\widehat{\deg(v)} := \frac{n-1}{|U|}|\{u \in U : \{u, v\} \in E(G)\}|$.

4. If $\widehat{\deg(v)} \geqslant (1-\varepsilon)\gamma(n-1)$, add v to D, and otherwise go to 2.

30

5. If $|D| = r$ output D, otherwise go to 2.

Let $W := \{v \in V(G) : \deg(v) \geqslant (1 - n^{-1/3})\gamma(n-1)\}$. Now we argue that W is relatively large. Indeed,

$$\gamma n(n-1) = \sum_{v \in V(G)} \deg(v) \leqslant (n-1)|W| + (n - |W|)(1 - n^{-1/3})\gamma(n-1),$$

from where dividing by $(n-1)$ and rearranging we get

$$|W| \geqslant n \cdot \frac{\gamma - (1 - n^{-1/3})\gamma}{1 - (1 - n^{-1/3})\gamma} = \frac{n^{2/3}\gamma}{1 - (1 - n^{-1/3})\gamma} \geqslant n^{2/3}\gamma.$$

Therefore, every time the algorithm enters step 2, the sampled vertex v will belong to $W\backslash D$ with probability at least $\mathbb{P}[v \in W] - \mathbb{P}[v \in D] \geqslant \gamma/\sqrt[3]{n} - \frac{r}{n} = \Omega(n^{-0.34})$, and thus the expected number of samples required to have sampled r distinct vertices from W is at most $O(rn^{0.34}) = O(n^{0.35})$. From this, if the algorithm enters step 2 at least $n^{0.36}$ times, then with high probability it will have sampled r distinct vertices from W.

To prove that the algorithm finishes correctly within the desired time, it suffices to prove two things:

> (**Completeness**) With high probability, every sampled vertex from W will pass the test of step 4, and thus be added to D.

> (**Soundness**) With high probability, no sampled vertex that passes the test of step 4 has degree less than $(1 - \varepsilon)^2\gamma(n-1)$.

Note that completeness implies the runtime to be $O(n^{0.36} \cdot \sqrt{n}) = o(n)$ w.h.p., since steps 3.–5. take time $O(\sqrt{n})$, and step 2 is entered at most $n^{0.36}$ times with high probability until $|D| = r$. On the other hand, soundness implies correctness of the algorithm, since if every vertex in D has degree at least $(1 - \varepsilon)^2\gamma(n-1)$, then we have

$$e(D, V\backslash D) = \left(\sum_{v \in D} \deg(v)\right) - 2e(D) \geqslant r(1 - \varepsilon)^2\gamma(n-1) - r^2$$

$$\geqslant (1 - \varepsilon)^2\gamma r(n - r) - r^2 = (1 - \varepsilon)^2\gamma r(n - r) \cdot \left(1 - \frac{r^2}{(1-\varepsilon)^2\gamma r(n-r)}\right)$$

$$\geqslant (1 - \lg^{-1/3}(n))^2 \left(1 - \frac{2r}{\gamma n}\right)\gamma r(n - r) \geqslant (1 - \lg^{-1/3}(n))^3\gamma r(n-r)$$

$$= (1 - O(\lg^{-1/3}(n)))\gamma r(n - r).$$

Now, note that $\mathbb{E}[\widehat{\deg(v)}] = \deg(v)$, and observe that by Hoeffding's inequality we have

$$\mathbb{P}[|\widehat{\deg(v)} - \deg(v)| \geqslant \varepsilon(1 - \varepsilon)\gamma(n-1)] = \mathbb{P}\left[\left|\widehat{\deg(v)}\frac{|U|}{n-1} - \deg(v)\frac{|U|}{n-1}\right| \geqslant \varepsilon(1-\varepsilon)\gamma|U|\right]$$

$$\leqslant 2\exp\left(\frac{-2(\varepsilon(1-\varepsilon)\gamma|U|)^2}{|U|}\right) \leqslant \exp(-\Omega(\sqrt[3]{n})),$$

where the last inequality used that $\gamma^{-1} = n^{o(1)}$ and also $\varepsilon^{-1} = n^{o(1)}$.

Therefore,

$$\mathbb{P}[\widehat{\deg(v)} \geqslant (1 - \varepsilon)\gamma(n-1) \mid \deg(v) < (1 - \varepsilon)^2\gamma(n-1)] \leqslant \mathbb{P}[|\widehat{\deg(v)} - \deg(v)| \geqslant \varepsilon(1 - \varepsilon)\gamma(n-1)]$$

$$\leqslant \exp(-\Omega(\sqrt[3]{n})),$$

and thus, after a union bound, with high probability every vertex v added to D in step 3 of the algorithm has degree at least $(1-\varepsilon)^2\gamma(n-1)$, which proves soundness.

Similarly, for completeness,

$$\mathbb{P}[\widehat{\deg(v)} < (1-\varepsilon)\gamma(n-1) \mid v \in W] = \mathbb{P}[\widehat{\deg(v)} < (1-\varepsilon)\gamma(n-1) \mid \deg(v) \geqslant (1-n^{-1/3})\gamma(n-1)]$$
$$\leqslant \mathbb{P}[|\widehat{\deg(v)} - \deg(v)| \geqslant (\varepsilon - n^{-1/3})\gamma(n-1)]$$
$$\leqslant \exp(-\Omega(\sqrt[3]{n})). \qquad \square$$

Theorem 9. *Given a graph with edge density γ such that $\max\{\gamma^{-1}, (1-\gamma)^{-1}\} = n^{o(1)}$, there is a randomized $O(\frac{n\lg n}{h_2(\gamma)}) = n^{1+o(1)}$ time algorithm that returns a biclique $K_{t,t}$ with $t = (1-o(1))\frac{\gamma\lg n}{h_2(\gamma)}$ with high probability. Furthermore, if we are allowed to query the degree of a vertex in $O(1)$ time, the algorithm can be made deterministic.*

Proof. Let $\varepsilon = \varepsilon(n)$ be an error term with $\varepsilon = \Theta(1/\sqrt[3]{\lg n})$, and let $r := \left\lfloor \frac{\lg n - 2\lg\lg n + \lg(h_2(\gamma)) + \lg(\gamma)}{h_2((1-\varepsilon)^2\gamma)} \right\rfloor$. Note that $r \sim \frac{\lg n}{h_2(\gamma)}$ since $h_2((1-\varepsilon)^2\gamma) \sim h_2(\gamma)$. We first find a subset $D \subseteq V(G)$ such that $|D| = r$ and the bipartite graph $G' := G[D, V\backslash D]$ has at least $(1-\varepsilon)\gamma|D| \cdot |V\backslash D|$ edges. In the deterministic model with $O(1)$ adjacency queries, we choose D to be the r vertices of highest degree in G. By sorting the vertices by degree, D can be computed in $O(n\lg n)$. We show in Lemma 37 that this guarantees at least $(1-\varepsilon)\gamma|D| \cdot |V\backslash D|$ edges. In the randomized setting, Lemma 38 gives a randomized $O(n)$ algorithm for constructing such a set D with high probability using random sampling.

Now, let $V^\star = \{v \in V\backslash D : d(v)/|D| \geqslant (1-\varepsilon)^2\gamma\}$, where $d(v)$ denotes the degree of v in G'. We can now lower bound the size of $V^\star$ by noting that

$$(1-\varepsilon)\gamma|D| \cdot |V\backslash D| \leqslant |E(G')| = \sum_{v\in V\backslash D} d(v) \leqslant |D| \cdot |V^\star| + |D|(1-\varepsilon)^2\gamma \cdot (|V\backslash D| - |V^\star|),$$

from where dividing by $|D|$ and rearranging we obtain

$$|V^\star| \geqslant |V\backslash D| \cdot \frac{(1-\varepsilon)\gamma - (1-\varepsilon)^2\gamma}{1-(1-\varepsilon)^2\gamma} = |V\backslash D| \cdot \frac{(1-\varepsilon)\varepsilon\gamma}{1-(1-\varepsilon)^2\gamma}$$
$$\geqslant |V\backslash D| \cdot \varepsilon \cdot (1-\varepsilon)\gamma \sim |V\backslash D| \cdot \varepsilon \cdot \gamma \geqslant \frac{n\gamma}{\lg n}.$$

Next, for each $v \in V^\star$, we compute an arbitrary subset $f(v) \subseteq N(v) \cap D$ of size $\lfloor (1-\varepsilon)^2\gamma|D| \rfloor$. By the pigeonhole principle, there is a subset $B \subseteq V$ such that $f(v) = f(w)$ for all $v, w \in B$ and

$$|B| \geqslant \frac{|V^\star|}{\binom{|D|}{\lfloor(1-\varepsilon)^2\gamma|D|\rfloor}} \geqslant \frac{|V^\star|}{2^{h_2((1-\varepsilon)^2\gamma)|D|}} \geqslant \frac{|V^\star|\lg^2(n)}{n\gamma h_2(\gamma)} \geqslant (1-o(1))\frac{\lg n}{h_2(\gamma)}.$$

Let $A := f(v)$ for any $v \in B$. Then, (A, B) is clearly a biclique, and it remains to observe that

$$|A| = \lfloor (1-\varepsilon)^2\gamma|D| \rfloor = (1-o(1))\frac{\gamma}{h_2(\gamma)}\lg n,$$

from where $\min\{|A|, |B|\} \geqslant (1-o(1))\frac{\gamma}{h_2(\gamma)}\lg n$.

After D has been chosen, the above construction can be performed in $O(\frac{n\lg n}{h_2(\gamma)})$ time deterministically as we show next. We initialize a list $V^\star$, and for every vertex $v \in V\backslash D$, as long

32

as $|V^\star| \leqslant \frac{n\gamma}{\lg n}$, we compute its subset $f(v)$ in time $O(|D|)$ by simply iterating over each vertex $u \in D$ and adding u to $f(v)$ if $\{u, v\} \in E(G)$, stopping when $|f(v)| = \lfloor(1-\varepsilon)^2\gamma|D|\rfloor$, in which case we add the pair $(v, f(v))$ to $V^\star$; this computation takes time $O(n \cdot |D|) = O(\frac{n \lg n}{h_2(\gamma)})$. At this point, it suffices to identify the $f(v^\star)$ that appears most frequently in $V^\star$, and output $(f(v^\star), \{v : (v, f(v^\star)) \in V^\star\})$. To identify such an $f(v^\star)$, we can simply sort all $f(v)$ lists lexicographically, in time $O(|V^\star|\lg|V^\star| \cdot |f(v)|) = O(n \lg n \frac{\gamma^2}{h_2(\gamma)})$, where the $|f(v)|$ factor is the cost of each comparison between lists, and then do a linear scan to find the most frequent. $\qquad\square$

7 Graphs with bounded shattering

This section shows how Theorem 11 can be strengthened for graphs in which the neighborhoods $\{N(v) : v \in V(G)\}$ form a structurally simple set-system, regardless of the edge density.

Given a set-system $\mathcal{F} \subseteq \mathcal{P}(X)$, define its *shatter function* $\pi_\mathcal{F} : \{1, \ldots, |X|\} \to \mathbb{N}$ by $\pi_\mathcal{F}(z) = \max_{X' \subseteq X, |X'|=z} |\{X' \cap F : F \in \mathcal{F}\}|$. Given a graph $G = (V, E)$, we define its shatter function π_G as the shatter function of the *neighborhood set-family* given by $\{N(v) : v \in V(G)\}$.

We are interested in graphs whose shatter function is polynomial. For instance, graphs whose neighborhood set-family has VC-dimension at most d satisfy $\pi_G(z) \leqslant \sum_{i=0}^{d} \binom{z}{i} = O(z^d)$, as a consequence of the Perles–Sauer–Shelah lemma [Sau72, She72].

An interesting subfamily of graphs with polynomial shatter functions is that of *d-dimensional semi-algebraic graphs with bounded complexity* (see, e.g. [CY25, Section 1.2.4] for precise definitions). Those graphs admit polynomial shatter functions, as a consequence of the Milnor–Thom theorem [FPS+17, Corollary 2.3]. This latter family includes many intersection graphs of geometric objects, e.g. graphs of intersections of disks in $\mathbb{R}^2$ (see e.g. [FPS+17, CY25] for other examples). Do [Do19] (see also [CY25, Theorem 5]) proved that d-dimensional semi-algebraic graphs with bounded complexity admit, for every $\varepsilon > 0$, biclique covers of weight $O_{d,\varepsilon}(n^{2-2/(d+1)+\varepsilon})$. Moreover, those biclique covers can be found efficiently [AAE+25]. A further subclass of graphs is that of *d-dimensional semi-linear graphs with bounded complexity* [CY25, Section 3] which includes, for example, interval graphs, and intersection graphs of axis-parallel boxes in $\mathbb{R}^d$. Cardinal and Yuditsky [CY25, Theorem 10] showed that semi-linear graphs satisfy $\mathsf{BC}(G) = \tilde{O}(n)$. We remark that all those families contain instances of dense graphs (with quadratic number of edges). Also, being semi-algebraic or semi-linear is a much stronger restriction than having a polynomial shatter function (see the discussion before Theorem 1.6 in [FPS19]).

In the next result we consider graphs with polynomial shatter function, and we can even bound $\mathsf{LBP}(G)$ in such graphs. The argument emulates that of Theorem 11, and it can also be implemented in deterministic $O(n^2)$-time as in Algorithm 1 (with the straightforward modifications).

Theorem 39. *Let G be an n-vertex graph with shatter function $\pi_G(z) \leqslant cz^d$, for some $c, d \geqslant 1$. Then $\mathsf{LBP}(G) = O(n^{1-1/(d+1)})$.*

Proof. Let $r = \lfloor n^{1/(d+1)} \rfloor$. Divide $V(G)$ into $t := \lceil n/r \rceil$ sets $V_1, \ldots, V_t$ of size as close as possible, and at most r. Let D be any tournament on $\{1, \ldots, t\}$. The edges completely contained inside some V_i form a biclique partition $\mathcal{B}_0$. For each V_i, consider the set $N_i = \bigcup_{j \in N_D(i)} V_j$. For each $S \subseteq V_i$, let $A_i(S) = \{v \in N_i : N_G(v) \cap V_i = S\}$. There are most $\pi_G(r)$ sets S for which $A_i(S)$ is non-empty. Considering $(S, A_i(S))$ for those sets, we get a biclique partition $\mathcal{B}_i$ of the bipartite graph between V_i and N_i. By construction the union over all $\mathcal{B}_i$ is a biclique partition of G.

33

Now we estimate, for an arbitrary $1 \leqslant i \leqslant t$ and $v \in V_i$, how many bicliques contain v. Clearly v is contained in at most r bicliques of $\mathcal{B}_0$. The bicliques from $\mathcal{B}_i$ contribute with at most $\pi_G(r)$; and for $j \neq i$ the vertex v can be in at most one biclique of $\mathcal{B}_j$. In total, we get $r + \pi_G(r) + t$. From the choice of r, this is at most $O(n^{1-1/(d+1)})$, as desired. $\qquad \square$

Observe that Theorem 39 combined with Lemma 36 immediately shows that every n-vertex $K_{t,t}$-free graph G with $\pi_G(z) \leqslant cz^d$ can have at most $O(tn^{2-1/(d+1)})$ edges. Fox, Pach, Sheffer, Suk, and Zahl [FPS+17, Theorem 2.1] showed that such graphs can have in fact at most $O_t(n^{2-1/d})$ edges, recently improved by Janzer and Pohoata [JP24] to $o_t(n^{2-1/d})$ edges if $k \geqslant d > 2$.

8 Dense random graphs

Chung, Erdős and Spencer [CES83] also asked for the average of $\mathrm{BC}(G)$ and $\mathrm{BP}(G)$ over all n-vertex graphs. Phrased in probabilistic terms, they asked for the expected value of $\mathrm{BC}(G)$ and $\mathrm{BP}(G)$ when G is drawn according to $G(n,p)$ with $p = 1/2$ (recall that $G(n,p)$ is the random binomial graph on n vertices where each edge is included independently with probability p). We answer their question in a more general form by considering all $p \in (0,1)$. We also consider the uniform random graph model $G(n,m)$, which is the model of random graphs chosen uniformly from $\mathcal{G}(n,m)$, the set of n-vertex graphs with m edges. In fact, we can also determine the 'typical' value of $\mathrm{BC}(G)$ and $\mathrm{BP}(G)$ for a dense random graph; from which the results on the expected value follow easily.

At a high level, we can get upper bounds in the $G(n,m)$ model immediately since there the edge density is deterministic, and thus Theorem 7 applies directly. The lower bounds are based on noting that the information-theoretic lower bounds also apply on average.

Theorem 40. *Let $p \in (0,1)$ be fixed. Given n, let $m = m(n) = \lfloor p\binom{n}{2} \rfloor$. Let G be a random graph chosen either according to $G(n,m)$ or $G(n,p)$. Then, with probability tending to 1 as n goes to infinity, we have $\mathrm{BC}(G) \sim \mathrm{BP}(G) \sim \frac{h_2(p)}{2} \cdot \frac{n^2}{\lg n}$.*

Proof. It suffices to prove the result for $G(n,m)$, as then the result is transferred automatically to the $G(n,p)$ model by the standard couplings between the two models, see e.g. [FK23, Theorem 1.4]. Let G be drawn uniformly from $\mathcal{G}(n,m)$. Note that the bound $\mathrm{BP}(G) \leqslant (h_2(p)/2 + o(1))n^2/\lg n$ follows deterministically from Theorem 7.

We need to prove that, with probability $1 - o(1)$, we also have $\mathrm{BC}(G) \geqslant (h_2(p)/2 - o(1))n^2/\lg n$. We know that each biclique cover of a graph G with weight $\mathrm{BC}(G)$ gives an encoding $f(G) \in \{0,1\}^\star$ of G using at most $\mathrm{BC}(G)(\lg n + O(1))$ bits. Since the encoding $f : \mathcal{G}(n,m) \to \{0,1\}^\star$ is injective, the set $\Gamma \subseteq \{0,1\}^\star$ of strings used by the encoding satisfies $|\Gamma| \geqslant |\mathcal{G}(n,m)|$. The number of strings in $\{0,1\}^\star$ of length at most k is $2^{k+1} - 1$. Hence, if k is such that $k = \lg |\Gamma| - \omega(1)$, then with high probability we have $\mathrm{BC}(G) \geqslant k/(\lg n + O(1))$. By the argument outlined at the beginning of Section 4 we have that $\lg |\Gamma| = (h_2(\gamma) + o(1))\binom{n}{2}$, so we can take $k = (h_2(\gamma) - o(1))\binom{n}{2}$ as well. This gives that with high probability we have $\mathrm{BC}(G) \geqslant (h_2(\gamma)/2 - o(1))n^2/\lg n$, as desired. $\qquad \square$

Corollary 41. *Let G be a random graph as in Theorem 40. Then $\mathbb{E}[BC(G)] \sim \mathbb{E}[BP(G)] \sim \frac{h_2(p)}{2} \cdot \frac{n^2}{\lg n}$.*

Proof. We consider $G \sim \mathcal{G}(n,m)$ in the uniform model first. Recall that $\mathrm{BP}(G) \leqslant (h_2(p)/2 + o(1))n^2/\lg n$ holds deterministically, so in particular $\mathbb{E}[\mathrm{BP}(G)] \leqslant (h_2(p)/2 + o(1))n^2/\lg n$ as well. Let $k = (h_2(p)/2 - o(1))n^2/\lg n$. Using Theorem 40, we have

$$\mathbb{E}[BC(G)] \geqslant k\,\mathbb{P}[BC(G) \geqslant k] = k(1 - o(1)),$$

which gives the desired lower bound. The proofs for G drawn from the binomial model $G(n, p)$ follow along the same lines, by noting that $|E(G)| = p\binom{n}{2} + o(n^2)$ with high probability (by, e.g. Chernoff inequalities). $\qquad\square$

9 Related work and applications

Biclique covers and partitions arise naturally in many areas of computer science, including the aforementioned applications to cryptography and computational geometry. We outline several more examples next.

Arguably, one of the most impactful applications of biclique partitions is to graph compression, kick-started by the seminal work of Feder and Motwani [FM95], which has been followed by empirically validated applications and theoretical improvements [CRGB25, HN13, FGK$^+$22].

A different line of work in which biclique coverings and partitions appear often is in circuit complexity (e.g., the *star complexity* of a graph), where these decompositions allow for constructing smaller circuits for graph-related problems [Juk13b, Juk12]. Biclique decompositions have been used to prove an upper bound on the size of monotone Boolean formulas for quadratic functions [Bub86] and to a minimization problem for Horn formulas [BDMT10]. Similarly, biclique coverings allow for reducing the size of CNF formulas in both graph and scheduling problems [Sub25].

Partitioning a graph G into bicliques $B_1 \sqcup B_2 \sqcup \ldots \sqcup B_k$ can also be understood from the adjacency matrix perspective, since then the adjacency matrix of G can be written as a sum of the adjacency matrices of the graphs B_i, appropriately padded with 0s. Due to the structure of adjacency matrices of bicliques, several linear-algebraic operations can be performed more efficiently over them, which has further motivated the study of biclique decompositions [Tuz84, Juk13a, FGK$^+$22].

The study of biclique decompositions has also arisen naturally in other areas of computer science such as automata theory, where it allowed Iván et al. to prove a separation result for the minimal number of states of nondeterministic finite automata (NFAs) using ε-transitions versus ε-free NFAs [ILN$^+$14].

The Erdős–Pyber theorem has been used to give a bound on the *local dimension* of posets [KMM$^+$20, DFGa$^+$21], and our results improve the constant factor in that bound (see [KMM$^+$20, Theorem 2]).

More generally, our work is part of a much broader field of research in discrete mathematics: *graph decompositions*. See [Sch22] for a recent survey on graph decomposition and [MPR95] for a survey about clique and biclique decompositions specifically.

10 Further directions

We have made significant progress in understanding the landscape of partite decompositions for graphs and uniform hypergraphs, showing tight results for both the total weight (Chung–Erdős–Spencer style) and for the maximum number of d-cliques each vertex belongs to (Erdős–Pyber style). Nonetheless, several avenues for future work remain open:

- We have shown how considering the edge density of graphs allows for better bounds when the density is bounded away from $1/2$ (Theorem 7). An interesting direction of future work is to study analogous density-aware results for d-uniform hypergraphs, with edge density $\gamma_d := |E(G)| / \binom{n}{d}$.

- We have shown in Section 6 how our techniques allow for efficiently finding large balanced bicliques in graphs. An interesting line of research is whether our results for hypergraphs can help find large balanced d-cliques in dense d-uniform hypergraphs. Concretely, d-cliques with each part having size $\Omega((\lg n)^{1/(d-1)})$ are ensured by [Erd64] (the hypergraph version of KST), and the best algorithmic results are given by a recent result of Espuña [Esp25]. In contrast, our hypergraph decomposition results (Theorems 14 and 18) yield very unbalanced d-cliques, in which all but 2 parts have size 1. Thus, an interesting question is whether similar decomposition results can be obtained with more balanced d-cliques.

- From the 'representation' perspective presented in Section 5, an interesting question is whether it is possible to efficiently support dynamic graphs, handling both edge insertions and deletions, as well as vertex insertions and deletions. The lazy deletion operation used for the densest subgraph approximation might be helpful as a starting point. In general terms, a more complete analysis of the complexity of common graph queries and operations under these biclique representations is a natural direction of future research.

References

[AAE⁺25]   Pankaj K. Agarwal, Boris Aronov, Esther Ezra, Matthew J. Katz, and Micha Sharir. Intersection queries for flat semi-algebraic objects in three dimensions and related problems. *ACM Trans. Algorithms*, 21(3), June 2025. 5, 33

[ACK21]   Sepehr Assadi, Deeparnab Chakrabarty, and Sanjeev Khanna. Graph Connectivity and Single Element Recovery via Linear and OR Queries. In Petra Mutzel, Rasmus Pagh, and Grzegorz Herman, editors, *29th Annual European Symposium on Algorithms (ESA 2021)*, volume 204 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 7:1–7:19, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. 25

[AEG⁺22]   Simon Apers, Yuval Efron, Pawel Gawrychowski, Troy Lee, Sagnik Mukhopadhyay, and Danupon Nanongkai. Cut Query Algorithms with Star Contraction . In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 507–518, Los Alamitos, CA, USA, November 2022. IEEE Computer Society. 25

[BDMT10]   Amitava Bhattacharya, Bhaskar DasGupta, Dhruv Mubayi, and György Turán. On approximate horn formula minimization. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide, and Paul G. Spirakis, editors, *Automata, Languages and Programming, 37th International Colloquium, ICALP 2010, Bordeaux, France, July 6-10, 2010, Proceedings, Part I*, volume 6198 of *Lecture Notes in Computer Science*, pages 438–450. Springer, 2010. 35

[BDSGV96]   Carlo Blundo, Alfredo De Santis, Luisa Gargano, and Ugo Vaccaro. On the information rate of secret sharing schemes. *Theoretical Computer Science*, 154(2):283–306, February 1996. 2

[Bei23]   Amos Beimel. Lower Bounds for Secret-Sharing Schemes for k-Hypergraphs. In Kai-Min Chung, editor, *4th Conference on Information-Theoretic Cryptography (ITC 2023)*, volume 267 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 16:1–16:13, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. 2

[BFM14]   Amos Beimel, Oriol Farràs, and Yuval Mintz. Secret-sharing schemes for very dense graphs. *Journal of Cryptology*, 29(2):336–362, December 2014. 2

[Bub86]   Siegfried Bublitz. Decomposition of graphs and monotone formula size of homogeneous functions. *Acta Inform.*, 23(6):689–696, 1986. 2, 4, 35

[BW09]   Nikhil Bansal and Ryan Williams. Regularity lemmas and combinatorial algorithms. In *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '09, page 745–754, USA, 2009. IEEE Computer Society. 4, 24

[CES83]   Fan Chung, Paul Erdős, and Joel Spencer. On the decomposition of graphs into complete bipartite subgraphs. In Paul Erdős, László Alpár, Gábor Halász, and András Sárközy, editors, *Studies in Pure Mathematics: To the Memory of Paul Turán*, pages 95–101. Birkhäuser, Basel, 1983. 1, 34

[Cha00]   Moses Charikar. *Greedy Approximation Algorithms for Finding Dense Components in a Graph*, page 84–95. Springer Berlin Heidelberg, 2000. 4, 25

[CLT15]   László Csirmaz, Péter Ligeti, and Gábor Tardos. Erdős-Pyber theorem for hypergraphs and secret sharing. *Graphs Combin.*, 31(5):1335–1346, 2015. 2, 3, 9, 11

[CRGB25]   Akshar Chavan, Sanaz Rabinia, Daniel Grosu, and Marco Brocanelli. A clique partitioning-based algorithm for graph compression, 2025. Arxiv preprint, `https://arxiv.org/abs/2502.02477`. 3, 4, 35

[CY25]   Jean Cardinal and Yelena Yuditsky. Compact Representation of Semilinear and Terrain-Like Graphs. In Anne Benoit, Haim Kaplan, Sebastian Wild, and Grzegorz Herman, editors, *33rd Annual European Symposium on Algorithms (ESA 2025)*, volume 351 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 67:1–67:19, Dagstuhl, Germany, 2025. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. 4, 5, 7, 29, 33

[DFGa$^+$21] Gábor Damásdi, Stefan Felsner, António Girão, Balázs Keszegh, David Lewis, Dániel T. Nagy, and Torsten Ueckerdt. On covering numbers, Young diagrams, and the local dimension of posets. *SIAM J. Discrete Math.*, 35(2):915–927, 2021. 35

[Do19]   Thao Do. Representation complexities of semialgebraic graphs. *SIAM J. Discrete Math.*, 33(4):1864–1877, 2019. 5, 33

[EP97]   Paul Erdős and László Pyber. Covering a graph by complete bipartite graphs. *Discrete Math.*, 170(1-3):249–251, 1997. 2

[Erd64]   Paul Erdős. On extremal problems of graphs and generalized graphs. *Israel J. Math.*, 2:183–190, 1964. 36

[Esp25]   Ferran Espuña. Finding partite hypergraphs efficiently, 2025. Arxiv preprint, `https://arxiv.org/abs/2508.10641`. 36

[FGK$^+$22] Alexandre P. Francisco, Travis Gagie, Dominik Köppl, Susana Ladra, and Gonzalo Navarro. Graph compression for adjacency-matrix multiplication. *SN Computer Science*, 3(3), March 2022. 4, 35

[FK23]   Alan Frieze and Michał Karoński. *Random graphs and networks: a first course*. Cambridge University Press, Cambridge, 2023. 34

[FM95]   Tomás Feder and Rajeev Motwani. Clique partitions, graph compression and speeding-up algorithms. *J. Comput. System Sci.*, 51(2):261–272, 1995. 3, 4, 35

[FM13]   Arash Farzan and J. Ian Munro. Succinct encoding of arbitrary graphs. *Theoretical Computer Science*, 513:38–52, 2013. 24

[FPS+17]   Jacob Fox, János Pach, Adam Sheffer, Andrew Suk, and Joshua Zahl. A semi-algebraic version of Zarankiewicz's problem. *J. Eur. Math. Soc. (JEMS)*, 19(6):1785–1810, 2017. 33, 34

[FPS19]   Jacob Fox, János Pach, and Andrew Suk. Erdős-Hajnal conjecture for graphs with bounded VC-dimension. *Discrete Comput. Geom.*, 61(4):809–829, 2019. 33

[FS13]   Zoltán Füredi and Miklós Simonovits. The history of degenerate (bipartite) extremal graph problems. In *Erdös centennial*, volume 25 of *Bolyai Soc. Math. Stud.*, pages 169–264. János Bolyai Math. Soc., Budapest, 2013. 28

[GMR06]   Alexander Golynski, J. Ian Munro, and S. Srinivasa Rao. Rank/select operations on large alphabets: a tool for text indexing. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, SODA '06, page 368–373, USA, 2006. Society for Industrial and Applied Mathematics. 24

[HN13]   Cecilia Hernández and Gonzalo Navarro. Compressed representations for web and social graphs. *Knowledge and Information Systems*, 40(2):279–313, April 2013. 4, 24, 35

[ILN+14]   Szabolcs Iván, Ádám Dániel Lelkes, Judit Nagy-György, Balázs Szörényi, and György Turán. Biclique coverings, rectifier networks and the cost of ε-removal. In Helmut Jürgensen, Juhani Karhumäki, and Alexander Okhotin, editors, *Descriptional Complexity of Formal Systems - 16th International Workshop, DCFS 2014, Turku, Finland, August 5-8, 2014. Proceedings*, volume 8614 of *Lecture Notes in Computer Science*, pages 174–185. Springer, 2014. 35

[JP24]   Oliver Janzer and Cosmin Pohoata. On the Zarankiewicz problem for graphs with bounded VC-dimension. *Combinatorica*, 44(4):839–848, 2024. 34

[Juk12]   Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*. Springer Berlin Heidelberg, 2012. 1, 35

[Juk13a]   Stasys Jukna. Complexity of linear boolean operators. *Foundations and Trends® in Theoretical Computer Science*, 9(1):1–123, 2013. 35

[Juk13b]   Stasys Jukna. Computational complexity of graphs. In *Advances in Network Complexity*, chapter 5, pages 99–153. John Wiley & Sons, Ltd, 2013. 35

[KMM+20]   Jinha Kim, Ryan R. Martin, Tomáš Masařík, Warren Shull, Heather C. Smith, Andrew Uzzell, and Zhiyu Wang. On difference graphs and the local dimension of posets. *European J. Combin.*, 86:103074, 13, 2020. 35

[KST54]   Thomas Kővári, Vera Sós, and Pál Turán. On a problem of K. Zarankiewicz. *Colloquium Mathematicae*, 3(1):50–57, 1954. 4, 28

[LMFB24]   Tommaso Lanciano, Atsushi Miyauchi, Adriano Fazzone, and Francesco Bonchi. A survey on the densest subgraph problem and its variants. *ACM Comput. Surv.*, 56(8), April 2024. 4, 25, 27

[Lot02]   M. Lothaire. *Algebraic combinatorics on words*, volume 90 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2002. A collective work by Jean Berstel, Dominique Perrin, Patrice Seebold, Julien Cassaigne, Aldo De Luca, Steffano Varricchio, Alain Lascoux, Bernard Leclerc, Jean-Yves Thibon, Veronique Bruyere, Christiane Frougny, Filippo Mignosi, Antonio Restivo, Christophe Reutenauer, Dominique Foata, Guo-Niu Han, Jacques Desarmenien, Volker Diekert, Tero Harju, Juhani Karhumaki and Wojciech Plandowski, With a preface by Berstel and Perrin. 12

[LSZ21]   Troy Lee, Miklos Santha, and Shengyu Zhang. Quantum algorithms for graph problems with cut queries. In *Proceedings of the Thirty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '21, page 939–958, USA, 2021. Society for Industrial and Applied Mathematics. 25

[Lup56]   Oleg Lupanov. On rectifier and switching-and-rectifier circuits. *Doklady Academii nauk SSSR*, 111, 1956. Available at https://web.vu.lt/mif/s.jukna/boolean/lupanov56.pdf, thanks to Stasys Jukna. 1, 8

[MPR95]   Sylvia D. Monson, Norman J. Pullman, and Rolf Rees. A survey of clique and biclique coverings and factorizations of $(0, 1)$-matrices. *Bull. Inst. Combin. Appl.*, 14:17–86, 1995. 35

[MT10]   Dhruv Mubayi and György Turán. Finding bipartite subgraphs efficiently. *Information Processing Letters*, 110(5):174–177, February 2010. 3, 4, 28, 29

[Nav16]   Gonzalo Navarro. *Compact Data Structures: A Practical Approach*. Cambridge University Press, September 2016. 23, 24

[Nec69]   Éduard Ivanovich Nechiporuk. The topological principles of self-correction. *Problemy Kibernet.*, (21):5–102, 1969. Available in Russian at https://web.vu.lt/mif/s.jukna/boolean/Russians/Nechiporuk-1969a.pdf, thanks to Stasys Jukna. 3, 16, 17

[Sau72]   Norbert W. Sauer. On the density of families of sets. *Journal of Combinatorial Theory, Series A*, 13(1):145–147, 1972. 33

[Sch22]   Stephan Schwartz. An overview of graph covering and partitioning. *Discrete Math.*, 345(8):Paper No. 112884, 17, 2022. 35

[She72]   Saharon Shelah. A combinatorial problem; stability and order for models and theories in infinitary languages. *Pacific Journal of Mathematics*, 41(1):247–261, April 1972. 33

[Sti94]   Douglas R. Stinson. Decomposition constructions for secret-sharing schemes. *IEEE Trans. Inf. Theory*, 40(1):118–125, 1994. 3

[Sub25]   Bernardo Subercaseaux. Asymptotically smaller encodings for graph problems and scheduling, 2025. (*ModRef workshop*), Arxiv preprint, https://arxiv.org/abs/2506.14042. 35

[Tuz84]   Zsolt Tuza. Covering of graphs by complete bipartite subgraphs: complexity of 0-1 matrices. *Combinatorica*, 4(1):111–116, 1984. 2, 4, 35

[Wil07]   Ryan Williams. Matrix-vector multiplication in sub-quadratic time: (some preprocessing required). In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, page 995–1001, USA, 2007. Society for Industrial and Applied Mathematics. 4, 24

[WW18]   Virginia Vassilevska Williams and R. Ryan Williams. Subcubic equivalences between path, matrix, and triangle problems. *J. ACM*, 65(5), August 2018. 4, 24

A Appendix

Proof of Claim 19. For the first item, note that r_d can be written as

$$r_d(n) = \frac{4^d}{d!} n^{d-1} \cdot a(n) \cdot b(d),$$

where b is an increasing function in d, and thus

$$r_d(n) = \frac{4n}{d} \cdot \frac{4^{d-1}}{(d-1)!} \cdot a(n) \cdot b(d) \geqslant \frac{4n}{d} \cdot \frac{4^{d-1}}{(d-1)!} \cdot a(n) \cdot b(d-1) = \frac{4n}{d} r_{d-1}(n).$$

The second item is direct from the fact that $\frac{n^{d-1} \lg \lg n}{\lg^2 n}$ is increasing in n, and so is $\frac{n^d}{\lg^3 n}$. For the third item, observe first that since $r_d(\cdot)$ is increasing by the previous item,

$$r_d(\lceil n/(d-1)\rceil) \leqslant r_d\left(\frac{n+d-2}{d-1}\right)$$

Then, note that $a(n) := \lg \lg n / \lg^2 n$ holds $a\left(\frac{n+d-2}{d-1}\right) \leqslant 1.01 a(n)$, for $n \geqslant 2^{2^{200d^2}}$. Indeed,

$$\frac{\lg \lg \left(\frac{n+d-2}{d-1}\right)}{\lg^2 \left(\frac{n+d-2}{d-1}\right)} \leqslant \frac{\lg \lg n}{\lg^2 \left(\frac{n+d-2}{d-1}\right)} \leqslant \frac{\lg \lg n}{\lg^2(n/(d-1))} = \frac{\lg \lg n}{(\lg n - \lg(d-1))^2},$$

but since $n \geqslant 2^{2^{200d^2}}$, we have $\lg n \geqslant 2^{200d^2} > 10000 \lg(d-1)$, and thus $\lg n - \lg(d-1) \geqslant 0.9999 \lg n$. We therefore have

$$a\left(\frac{n+d-2}{d-1}\right) \leqslant \frac{\lg \lg n}{(0.9999 \lg n)^2} \leqslant 1.01 a(n).$$

Now note that as $n \geqslant 2^{2^{200d^2}}$,

$$\left(\frac{n+d-2}{n}\right)^{d-1} = \left(1 + \frac{d-2}{n}\right)^{d-1} \leqslant \exp((d-2)(d-1)/n) \leqslant 1.01,$$

from where we conclude

$$r_d(\lceil n/(d-1)\rceil) \leqslant \frac{4^d}{d!}(n+d-2)^{d-1} \cdot \frac{1}{(d-1)^{d-1}} \cdot 1.01 a(n) \cdot b(d)$$

$$\leqslant 1.01^2 \cdot \frac{1}{(d-1)^{d-1}} \frac{4^d}{d!} n^{d-1} a(n) b(d) \leqslant 1.03 \frac{r_d(n)}{(d-1)^{d-1}}. \qquad \square$$

Proof of Claim 21. The number of auxiliary d-cliques containing v is at most

$$(\star) := \sum_{\substack{\vec{x}\in\mathcal{S}_d \\ x_i\neq 0}} \binom{|P_i|}{x_i-1} \cdot \left(\prod_{j\in[d-1]\setminus\{i,f(\vec{x})\}} \binom{|P_j|}{x_j}\right) h_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|)$$

$$\leqslant \sum_{\substack{\vec{x}\in\mathcal{S}_d \\ x_i\neq 0}} \frac{|P_i|^{x_i-1}}{(x_i-1)!} \cdot \left(\prod_{j\in[d-1]\setminus\{i,f(\vec{x})\}} \frac{|P_j|^{x_j}}{x_j!}\right) h_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|)$$

$$\leqslant \sum_{\substack{\vec{x}\in\mathcal{S}_d \\ x_i\neq 0}} x_i \left(\prod_{j\in[d-1]\setminus\{f(\vec{x})\}} \frac{1}{x_j!}\right)\left(|P_i|^{x_i-1}\prod_{j\in[d-1]\setminus\{i,f(\vec{x})\}} |P_j|^{x_j}\right) h_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|)$$

$$= \sum_{\substack{\vec{x}\in\mathcal{S}_d \\ x_i\neq 0}} x_i \left(\prod_{j\in[d-1]\setminus\{f(\vec{x})\}} \frac{1}{x_j!}\right)\left(|P_i|^{x_i-1}\prod_{j\in[d-1]\setminus\{i,f(\vec{x})\}} |P_j|^{x_j}\right) c(d)\cdot\frac{|P_{f(\vec{x})}|^{x_{f(\vec{x})}}}{\lg^3|P_{f(\vec{x})}|},$$

where the last equality introduced notation $c(d) := 2^{d2^{200d^2}}$. We now observe that

$$(\star) = \sum_{\substack{\vec{x}\in\mathcal{S}_d \\ x_i\neq 0}} x_i \left(\prod_{j\in[d-1]\setminus\{f(\vec{x})\}} \frac{1}{x_j!}\right)\left(|P_i|^{x_i-1}\prod_{j\in[d-1]\setminus\{i\}} |P_j|^{x_j}\right)\frac{c(d)}{\lg^3|P_{f(\vec{x})}|}$$

$$\leqslant \sum_{\substack{\vec{x}\in\mathcal{S}_d \\ x_i\neq 0}} x_i x_{f(\vec{x})}! \left(\prod_{j\in[d-1]} \frac{1}{x_j!}\right)[n/(d-1)]^{d-1}\frac{c(d)}{\lg^3(n/d)} \qquad \text{(Using } |P_{f(\vec{x})}| \geqslant \lfloor n/(d-1)\rfloor \geqslant n/d\text{)}$$

$$\leqslant d\cdot(d-1)^d[n/(d-1)]^{d-1}\frac{c(d)}{(\lg n - \lg d)^3}. \qquad\qquad\qquad\qquad\qquad \text{(By Equation (2))}$$

We now need once again the fact

$$[n/(d-1)]^{d-1} \leqslant \left(\frac{n+d-2}{d-1}\right)^{d-1} = \left(\frac{n}{d-1}\right)^{d-1}\left(1+\frac{d-2}{n}\right)^{d-1}$$

$$\leqslant \left(\frac{n}{d-1}\right)^{d-1} e^{(d-2)(d-1)/n} \leqslant 1.01\left(\frac{n}{d-1}\right)^{d-1}.$$

From this, we have

$$(\star) \leqslant 1.01 d\cdot n^{d-1}\frac{c(d)}{(\lg n - \lg d)^3} \leqslant 2d\cdot n^{d-1}\frac{c(d)}{\lg^3 n}, \qquad\qquad \text{(Using } \lg d \leqslant 0.1\lg n\text{)}$$

and as $r_d(n) = \frac{4^d}{d!}\cdot c(d)\cdot\frac{n^{d-1}\lg\lg n}{\lg^2 n}$, we have $(\star)/r_d(n) \leqslant \frac{2d\cdot d!}{4^d\lg n\lg\lg n} \leqslant \frac{2d\cdot d!}{\lg n} < \frac{1}{3}$, where the last inequality holds since $n \geqslant 2^{2^{200d^2}}$ implies $\lg n \geqslant 2^{200d^2}$, and $d! \leqslant d^d = 2^{d\lg d}$. $\qquad\square$

Proof of Claim 22. We start by noting that

$$|\mathcal{C}| \leqslant \sum_{\vec{x} \in \mathcal{S}_d} \left(\prod_{i \in [d-1] \setminus \{f(\vec{x})\}} \binom{|P_i|}{x_i} \right) h_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|)$$

$$\leqslant \sum_{\vec{x} \in \mathcal{S}_d} \left(\prod_{i \in [d-1] \setminus \{f(\vec{x})\}} \frac{|P_i|^{x_i}}{x_i!} \right) h_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|).$$

Now, we split the sum according to whether some $x_i = d$ or not. Thus,

$$|\mathcal{C}| \leqslant \left[\sum_{\substack{\vec{x} \in \mathcal{S}_d \\ d \notin \vec{x}}} \left(\prod_{i \in [d-1] \setminus \{f(\vec{x})\}} \frac{|P_i|^{x_i}}{x_i!} \right) h_{x_{f(\vec{x})}}(|P_{f(\vec{x})}|) \right] + (d-1)h_d(\lceil n/(d-1) \rceil)$$

$$\leqslant \left[\sum_{\substack{\vec{x} \in \mathcal{S}_d \\ d \notin \vec{x}}} \left(\prod_{i \in [d-1] \setminus \{f(\vec{x})\}} \frac{|P_i|^{x_i}}{x_i!} \right) \frac{c(x_{f(\vec{x})})|P_{f(\vec{x})}|^{x_{f(\vec{x})}}}{\lg^3 |P_{f(\vec{x})}|} \right] + (d-1)h_d(\lceil n/(d-1) \rceil)$$

$$\leqslant \left[\sum_{\substack{\vec{x} \in \mathcal{S}_d \\ d \notin \vec{x}}} \lceil n/(d-1) \rceil^d x_{f(\vec{x})}! \left(\prod_{i \in [d-1]} \frac{1}{x_i!} \right) \frac{c(d-1)}{\lg^3 |P_{f(\vec{x})}|} \right] + (d-1)h_d(\lceil n/(d-1) \rceil)$$

$$\leqslant \lceil n/(d-1) \rceil^d d! \cdot \frac{(d-1)^d}{d!} \frac{c(d-1)}{\lg^3(n/d)} + (d-1)h_d(\lceil n/(d-1) \rceil)$$

$$\leqslant 1.01 \frac{n^d \cdot c(d-1)}{\lg^3 n} + \frac{1.01}{(d-1)^{d-1}} \frac{c(d)n^d}{\lg^3 n}$$

$$= h_d(n) \cdot \left(\frac{1.01c(d-1)}{c(d)} + \frac{1.01}{(d-1)^{d-1}} \right) < h_d(n).$$

where the second-to-last inequality used the same analysis as in the proof of Claim 21, and the last inequality used, e.g., $\frac{c(d-1)}{c(d)} < 1/10$ for $d \geqslant 3$. $\qquad\square$