Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 5 (2022)

Sunflowers: from soil to oil *

Anup Rao
University of Washington
anuprao@cs.washington.edu

Abstract

A *sunflower* is a collection of sets whose pairwise intersections are identical. In this article, we shall go sunflower-picking. We find sunflowers in several seemingly unrelated fields, before turning to discuss recent progress on the famous sunflower conjecture of Erdős and Rado, made by Alweiss, Lovett, Wu and Zhang, as well as a related resolution of the *threshold vs expectation threshold conjecture* of Kahn and Kalai discovered by Park and Pham. We give short proofs for both of these results.

1 Sunflowers

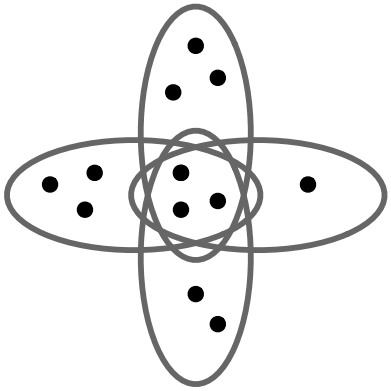

Figure 1: A sunflower with 4 petals.

The moral of Ramsey theory is that large systems can exhibit surprising structure. There are many examples of this kind, starting with the prototypical one: every graph on n vertices either contains a clique[1] on $(1/2) \cdot \log_2 n$ vertices, or an independent set[2]

*This exposition appears as a companion piece to a talk given at the current events bulletin at the Joint Mathematical Meeting of the AMS in 2022.

[1]Mutually adjacent vertices

[2]Mutually non-adjacent vertices

ISSN 1433-8092

on $(1/2) \cdot \log_2 n$ vertices. Roth's theorem [15] proves that every subset of $\{1, \ldots, n\}$ of density $\Omega(1)$ must contain an arithmetic progression[3]. The Hales-Jewett theorem [9] and Ajtai and Szemeredi's Corner theorem [1] are other examples of this phenomenon.

A *sunflower* with w petals is a collection of w sets whose pairwise intersections are identical. The common intersection is called the *core*. In 1960, Erdős and Rado [4] proved a Ramsey theoretic result concerning sunflowers: every large collection of sets must contain a sunflower. They gave a simple inductive argument showing that every collection of more than $k! \cdot (w-1)^k$ sets of size at most k must contain a sunflower with w petals[4]. There are examples with $\Omega(w)^k$ sets that have no sunflowers, and they conjectured that the correct bound is $O(w)^k$.

The seeds were planted, and the search for sunflowers and sunflower lemmas began in earnest. We begin this article by taking a tour through various fields where sunflowers are essential. We shall see examples relevant to finding arithmetic progressions in sumsets, understanding models of computation such as monotone boolean circuits and data structures, and fundamental questions about the threshold of a monotone function. In each of these arenas, we skip details and zoom in to focus on the role played by sunflowers.

In 2019, Alweiss, Lovett, Wu and Zhang [2] made significant progress towards proving the sunflower conjecture. Subsequent refining by myself [13], Frankston, Kahn, Narayanan and Park [7] and Bell, Chueluecha and Warnke [3] led to the result that every collection of $O(w \log k)^k$ sets of size at most k must contain a sunflower with w petals. A few months later, some of these ideas were used by Park and Pham [11] to give a surprisingly simple and elegant resolution of the threshold vs expectation threshold conjecture of Kahn and Kalai [10]. In this article, we present a version of these arguments that give the easiest proofs yet. In fact, we give a single argument that simultaneously proves the sunflower bound and resolves the conjecture about thresholds.

2 Arithmetic Progressions in Sumsets

In 1992, Erdős and Sárközy [5] used sunflowers to find arithmetic progressions in subset sums. Given a set $T \subseteq \{1, \ldots, n\}$, let $\mathsf{sum}(T)$ denote the quantity $\sum_{x \in T} x$. Then they proved:

Theorem 1 ([5]). Given any set $S \subseteq \{1, \ldots, n\}$ of size $|S| \gg \log^2 n$, there are subsets $T_1, \ldots, T_{w+1} \subseteq S$, with $w \approx |S|/\log^2 n$, such that the sequence $\mathsf{sum}(T_1), \cdots, \mathsf{sum}(T_{w+1})$ is an arithmetic progression.

Much like the sunflower lemma, this is an example of finding structure in a large system. However, the structure we seek here is an arithmetic progression; what does this have to do with sunflowers? Erdős and Sárközy move between the two structures as follows. First, by counting the number of possible sums that can be obtained by

[3] Three numbers $a, a+d, a+2d$

[4] Often the sunflower lemma is stated under the assumption that each set is of size *exactly* k rather than at most k. Here we use the more general form because many application rely on this form, and all of the ideas for proving the lemmas carry through.

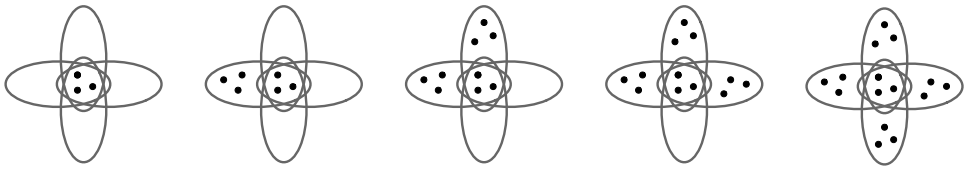

Figure 2: 4 petals induces an arithmetic progression of length 5.

subsets of S, and estimating a binomial coefficient, they show that some $(w \log n)^{\log n}$ subsets of S of size $\log n$ must attain the same sum. By the sunflower lemma, and the choice of parameters, this collection of sets is guaranteed to contain a sunflower. The proof is completed by the following claim, whose proof we leave as an exercise (Figure 2):

Claim 2. If $S_1, \ldots, S_w \subseteq \{1, \ldots, n\}$ is a sunflower with core C, $|S_1| = \cdots = |S_w|$, and $\mathsf{sum}(S_1) = \mathsf{sum}(S_2) = \cdots = \mathsf{sum}(S_w)$, then

$$\mathsf{sum}(C), \mathsf{sum}(S_1), \mathsf{sum}(S_1 \cup S_2), \ldots, \mathsf{sum}(S_1 \cup \cdots \cup S_w)$$

is an arithmetic progression.

3 Monotone Circuit Lower Bounds

Sunflowers have had a huge impact in theoretical computer science. Perhaps the most well-known example is Razborov's [14] proof from 1985 that there are no small monotone circuits computing the *clique* function. Here, we give a cartoon description of this clever argument.

A boolean circuit computes with the help of *gates* implementing boolean logic. These logic gates can compute the OR, AND or negation of their inputs. The inputs to the gates are either the outputs of other gates, or input variables. The size of the circuit is the number of wires used, which is the same as the number of connections made between gates. A *monotone* circuit is a boolean circuit that does not have any gates computing negations. The circuit computs a function if there is a gate whose value is equal to the value of the function, for every choice of the input variables.

For a graph G on n vertices, and a set S of vertices, define

$$\mathsf{clique}_S(G) = \begin{cases} 1 & \text{if } G \text{ contains a clique on the vertices of } S, \\ 0 & \text{otherwise.} \end{cases}$$

The function of interest for us is

$$\mathsf{clique}_k(G) = \bigvee_{S \subseteq \{1,\ldots,n\}, |S|=k} \mathsf{clique}_S(G),$$

which computes whether or not the graph contains a clique of size k. Razborov's argument leads to the following theorem:

Theorem 3 ([14]). For every $\epsilon > 0$, and n large enough, every monotone circuit computing $\mathsf{clique}_{n^{1/3-\epsilon}}(G)$ must have size at least $2^{\Omega(n^{1/6-\epsilon})}$.

Razborov proves that this function requires exponentially large monotone circuits, if $k \approx n^{1/3}$. Razborov's result is one of the few examples where we are able to prove lower bounds on reasonable models of computation: it is a gem of theoretical computer science.

At a high level, sunflowers are used critically to show that any circuit computing clique_k can be used to obtain a smaller circuit with the same ability. Each such step involves a tiny error. We obtain a good approximation to the original circuit that is so simple that we can directly reason that it does not work. This proves that the original circuit does not work either.

Now, let us give a few more details. Let G be a graph on n vertices that contains a uniformly random clique of size k, and no other edges. Let H be a uniformly random $(k-1)$-partite graph. G always contains a clique of size k, while H never contains a clique of size k. A monotone circuit computing the clique function would have to output 1 on G and 0 on H. An input variable to the circuit is the indicator for the presence of an edge, which can be thought of as clique_S for some set S of size 2.

Let us discuss how to approximate the circuit by a simpler circuit. First, we claim that $\mathsf{clique}_S \wedge \mathsf{clique}_T$ can be safely replaced by $\mathsf{clique}_{S\cup T}$. This is because by the choice of G,

$$\mathsf{clique}_S(G) \wedge \mathsf{clique}_T(G) \leq \mathsf{clique}_{S\cup T}(G),$$

and by the choice of H,

$$\mathsf{clique}_S(H) \wedge \mathsf{clique}_T(H) \geq \mathsf{clique}_{S\cup T}(H).$$

Thus, carrying out this approximation preserves the ability of the circuit to distinguish G from H, while reducing the size of the circuit.

Sunflowers play a key role in approximating OR gates, via the following claim:

Claim 4. If $S_1, \ldots, S_w$ form a sunflower with core C, and all sets S_i are of size at most $\sqrt{k}$, then

$$\mathsf{clique}_{S_1}(G) \vee \cdots \vee \mathsf{clique}_{S_w}(G) \leq \mathsf{clique}_C(G),$$

and with high probability over the choice of H,

$$\mathsf{clique}_{S_1}(H) \vee \cdots \vee \mathsf{clique}_{S_w}(H) \geq \mathsf{clique}_C(H).$$

When the input is G, the claim is trivial. When the input is H, the approximation causes a problem if there is a clique on C in H, yet none of the petals constitute a clique. This is extremely unlikely to happen: given that the core is a clique, the events that the petals are also cliques are independent, and the choice of parameters ensures that each occurs with probability $\Omega(1)$. So, one can argue that one of the petals will be a clique with probability $1 - 2^{-\Omega(w)}$.

Thus, if t is large enough, any expression of the type

$$\mathsf{clique}_{S_1} \vee \ldots \vee \mathsf{clique}_{S_t}$$

can be approximated by a smaller expression of the same type—use the sunflower lemma to find a sunflower among the sets and replace it by the core. Repeatedly applying these operations, one can show that any arbitrary small monotone circuit can be approximated by a circuit whose structure is so simple that it is trivial to verify that it cannot distinguish G from H.

4 Lower Bounds for Data Structures

Data structures are a fundamental concept in computer science. They are used to efficiently maintain an object so that the object can be quickly modified and queried. Our next example is a lower bound on the running time of data structures for the problem of maintaining a set and computing its minimum, from my recent work with Ramamoorthy [12], building on ideas from [6, 8]. Our work shows:

Theorem 5 ([12]). Any non-adaptive data structure of word size $\log n$ that allows to add or delete elements from a subset of $\{1, \ldots, n\}$ and compute the minimum element of the set must access $\Omega(\log n / \log \log n)$ locations for some operation.

The result is independent of the algorithm used to implement the data structure and the particular encoding of the data (namely T) used, the argument only relies on the sets of locations that the data structure reads and writes to. A valid data structure for our purposes is one that encodes the set T as a vector $\mathsf{enc}(T) \in \{1, \ldots, n\}^m$. The data structure is associated with a family of subsets of the coordinates of the encoding $S_1, \ldots, S_n \subseteq \{1, \ldots, m\}$ and an algorithm for manipulating $\mathsf{enc}(T)$. For each i, the algorithm is able change $\mathsf{enc}(T)$ to either $\mathsf{enc}(T \cup \{i\})$ or $\mathsf{enc}(T - \{i\})$ and compute the new minimum of the set by reading and writing to the coordinates of $\mathsf{enc}(T)$ given by S_i. Under just these assumptions, the argument proves that some set S_i must be of size $\Omega(\log n / \log \log n)$.

If all of the sets $S_1, \ldots, S_n$ are of size $\ll \frac{\log n}{\log \log n}$, the choice of parameters implies that there is a sunflower, say $S_1, \ldots, S_w$, with $w \approx (\log n)^{100}$, and core C. Then the key claim is:

Claim 6. If $S_1, \ldots, S_w$ is a sunflower with core C, then every subset of $T \subseteq \{1, \ldots, w\}$ has an encoding as a vector $\mathsf{enc}(T) \in \{1, \ldots, n\}^{|C|}$.

This claim combined with a straightforward counting argument implies that $|C| \geq \Omega(\frac{\log n}{\log \log n})$, proving that one of the sets S_i must be large. To prove the claim, for any set T, arrive at its encoding by deleting the elements of the set $\{1, \ldots, w\} - T$ from the encoding of $\{1, \ldots, w\}$. The claimed encoding corresponds to the contents of the core at this point, which is a string in $\{1, \ldots, n\}^{|C|}$. T can be recovered from the encoding by computing the minimum of T, then deleting the minimum, then computing the minimum and deleting it, and repeating these operations over and over until the entire set T has been recovered. Because each of these operations only interact on the coordinates of $\mathsf{enc}(T)$ that correspond to the core, the contents of the core are enough to simulate the entire process and determine T.

5 Estimating the Threshold of Monotone Functions

Suppose we sample a random graph by including each edge independently with probability ϵ. How can we estimate the probability that the graph contains a perfect matching?

This is a special case of a more general question. Let $2^{[n]}$ denote the set of subsets of $\{1, \dots, n\}$, and let $f : 2^{[n]} \to \{0, 1\}$ be a *monotone* function, meaning that $X \subseteq Y$ implies that $f(Y) \geq f(X)$. Let $\mathbf{X} \in 2^{[n]}$ be sampled by including each $i \in \mathbf{X}$ independently with probability ϵ. Because f is monotone, $\mathbb{E}[f(\mathbf{X})]$ is increasing in ϵ. The *threshold* of f is the value of ϵ for which $\mathbb{E}[f(\mathbf{X})] = 1/2$. There are a couple of generic ways to bound $\mathbb{E}[f(\mathbf{X})]$, and these bounds induce other kinds of thresholds that capture something about the structure of f. These ideas were explored extensively by Kahn and Kalai [10], Talagrand [16], Frankston, Kahn, Narayanan and Park [7] and Park and Pham [11].

Given a family of sets $\mathcal{F}$ and a set X, define the *shadow* $\mathcal{F}_X = \{F \in \mathcal{F} : F \subseteq X\}$. It is easy to see that every monotone function f admits a minimal collection of sets $\mathcal{F}$ such that $f(X) = 1$ for $X \in \mathcal{F}$. Moreover, if $\mathcal{F}$ is the minimal family for f, then $f(X) = 1$ if and only if $|\mathcal{F}_X| \geq 1$. So, by the union bound:

$$\mathbb{E}[f(\mathbf{X})] \leq \sum_{Y \in \mathcal{F}} \mathbb{P}[Y \subseteq \mathbf{X}] = \mathbb{E}[|\mathcal{F}_X|]. \tag{5.1}$$

More generally, for every monotone function g with $f \leq g$, meaning that $f(X) \leq g(X)$ for all X, we have the bound:

$$\mathbb{E}[f(\mathbf{X})] \leq \mathbb{E}[\mathcal{G}_{\mathbf{X}}|], \tag{5.2}$$

where here $\mathcal{G}$ is the family of minimal sets of g.

The *expectation-threshold* of f is the largest value of ϵ for which the right-hand-side of (5.2) is equal to $1/2$ for some monotone g with $f \leq g$. By (5.2), the threshold is always at least the expectation-threshold. When f computes whether or not a graph has a perfect matching, the threshold is $\approx \frac{\log n}{n}$, while the expectation threshold is $\approx \frac{1}{n}$. Kahn and Kalai conjectured that this is the worst possible ratio: the threshold is always at most $O(\log n)$ times larger than the expectation-threshold.

In general, the above union bound can be quite far from tight. It is not tight when the events $Y \subseteq \mathbf{X}$ have intersections of significant measure. There is a more sophisticated way to get upper bounds on $\mathbb{E}[f(\mathbf{X})]$, as observed by Talagrand [16]—it can be thought of as a fractional variant of the union bound discussed above. Suppose there is a probability distribution $\mathbf{Z}$ on $2^{[n]}$ and κ satisfying

$$f(U) \leq \kappa \cdot \mathbb{E}[1_{\mathbf{Z} \subseteq U} \cdot \epsilon^{-|\mathbf{Z}|}],$$

for all sets U. Then we obtain the upper bound:

$$\mathbb{E}[f(\mathbf{X})] \leq \kappa \cdot \mathbb{E}[1_{\mathbf{Z} \subseteq \mathbf{X}} \cdot \epsilon^{-|\mathbf{Z}|}] = \kappa, \tag{5.3}$$

since for any fixed Z, the probability that $Z \subseteq \mathbf{X}$ is exactly $\epsilon^{|Z|}$.

The *fractional-expectation-threshold* is the largest value of ϵ for which there is a $\mathbf{Z}$ satisfying the above condition with $\kappa = 1/2$. The union bound of (5.2) can also be proved using (5.3), because if $f \leq g$ and $\mathcal{G}$ is the set of minimal sets of g, then we can sample $\mathbf{Z}$ so that

$$\mathbb{P}[\mathbf{Z} = Z] = \begin{cases} \frac{\mathbb{E}[1_{Z \subseteq \mathbf{X}}]}{\mathbb{E}[|\mathcal{G}_{\mathbf{X}}|]} & \text{if } Z \in \mathcal{G}, \\ 0 & \text{otherwise,} \end{cases}$$

then because $\mathbb{E}[1_{Z \subseteq \mathbf{X}}] = \epsilon^{|Z|}$, we have that for any U

$$f(U) \leq g(U) \leq |\mathcal{G}_U| \leq \mathbb{E}[|\mathcal{G}_{\mathbf{X}}|] \cdot \mathbb{E}[1_{\mathbf{Z} \subseteq U} \cdot \epsilon^{-|\mathbf{Z}|}],$$

proving (5.2). So, the bound given by (5.3) is certainly at least as good as the bound given by (5.2). In particular, this implies that the threshold is at least as large as the fractional-expectation-threshold, which in turn is at least the expectation-threshold. But how far apart can these numbers be?

Talagrand conjectured that the fractional-expectation-threshold is within a multiplicative factor of $O(\log n)$ from the threshold, and within an $O(1)$ factor of the expectation-threshold. Following the recent progress on the sunflower lemma, Frankston, Kahn, Narayanan and Park [7] proved that the fractional-expectation-threshold is within $O(\log n)$ of the threshold, so resolving Talagrand's first conjecture. Subsequently, Park and Pham proved that the expectation threshold is within a $O(\log n)$ factor of the threshold, resolving Kahn and Kalai's conjecture:

Theorem 7 ([11]). For any monotone function $f : \{0,1\}^n \to \{0,1\}$, the threshold is at most $O(\log n)$ times larger than the expectation threshold.

This allows to compute the threshold for many graph properties, such as perfect matchings, Hamiltonian circuits and bounded degree spanning trees. The ideas used to prove new sunflower lemmas play a key role in these proofs.

Talagrand made an important observation that suggests a definition that is ultimately used to prove the improved sunflower lemma. Suppose that κ is the smallest number for which there is a $\mathbf{Z}$ establishing (5.3). Then by von-Neumann's minimax theorem, there is a distribution on $\mathbf{U}$ such that for every choice of Z,

$$\mathbb{E}[f(\mathbf{U})] \geq \kappa \cdot \mathbb{E}[1_{Z \subseteq \mathbf{U}} \cdot \epsilon^{-|Z|}]. \tag{5.4}$$

Without loss of generality, we may assume that $\mathbf{U}$ is supported on the minimal sets of f, since we can always modify the distribution in this way and preserve the inequality. So, after making this change, we can rewrite (5.4) as:

$$\epsilon^{|Z|}/\kappa \geq \mathbb{E}[1_{Z \subseteq \mathbf{U}}]. \tag{5.5}$$

This shows that $\mathbf{U}$ has a very interesting property: it is *spread*, in the sense that it is unlikely to contain any fixed set Z: $\mathbb{P}[Z \subseteq \mathbf{U}] \leq \epsilon^{|Z|}/\kappa \leq r^{-|Z|}$, for $r = \kappa/\epsilon$.

The ideas used to prove the sunflower lemma are ultimately useful to prove the threshold vs expectation threshold conjecture, as well as the threshold vs fractional expectation threshold conjecture. Let us briefly put on hold our study of these thresholds to discuss how the concept of being spread is useful to prove the new sunflower bound.

6 Sunflowers in Spread Families

At last we return to the sunflower question: how many sets of size k are sufficient to ensure the presence of a sunflower with w petals? Alweiss, Lovett, Wu and Zhang discovered an elementary counting argument that is surprisingly powerful to help answer this question.

Given a collection $\mathcal{S}$ of sets, let $\mathbf{U} \in \mathcal{S}$ be uniformly random. As in the last section, we shall say that $\mathbf{U}$ is *r-spread* (for some parameter $r = O(w \log k)$) if for every set Z, $\mathbb{P}[Z \subseteq \mathbf{U}] \leq r^{-|Z|}$. Now, suppose $|\mathcal{S}| \geq r^k$. If $\mathbf{U}$ is not r-spread, then there is a set Z such that the family $\mathcal{S}' = \{S \in \mathcal{S} : Z \subseteq S\}$ has at least $r^{k-|Z|}$ sets. In this case, we inductively find a sunflower in the family of sets of size at most $k - |Z|$ obtained by deleting Z from the sets of $\mathcal{S}'$. Adding Z back into this sunflower gives us a sunflower in our original family of sets. So, it only remains to find sunflowers in spread families.

We prove the following claim in the next section. (All logarithms are computed base 2).

Claim 8. For $r = (64 \log k)/\epsilon$, if $\mathcal{S}$ is such that $\mathbf{U}$ is r-spread, and $\mathbf{W}$ is a uniformly random set of size ϵn, then $\mathbb{P}[\mathcal{S}_{\mathbf{W}} = \emptyset] \leq 1/2$.

Assuming the claim, let $\mathbf{W}_1, \ldots, \mathbf{W}_{2w}$ be a random partition of the universe into $2w$ sets, and set $\epsilon = 1/(2w)$, so $r = 128w \log k$. Claim 8 implies that at least w of these sets will contain a set of the family in expectation, and so there must be w mutually disjoint sets: a sunflower with w petals.

7 A Clever Counting Argument

Finally we arrive at the key technical theorem which will help us prove the new sunflower bound as well as resolve the threshold vs expectation threshold conjecture. Recall that we defined the shadow $\mathcal{G}_U = \{Y \in \mathcal{G} : Y \subseteq U\}$, and $\mathbf{U}$ is r-spread if $\mathbb{P}[Z \subseteq U] \leq r^{-|Z|}$. We prove:

Theorem 9. Let $\mathcal{S} \subseteq 2^{[n]}$ be a family of sets of size at most k. Then there is a distribution on pairs $(\mathbf{W}, \mathcal{G})$, where $\mathbf{W} \in 2^{[n]}$ is a uniformly random set of size ϵn and $\mathcal{G} \subseteq 2^{[n]}$ is a family of sets, such that either $\mathcal{S}_{\mathbf{W}} \neq \emptyset$, or for every $S \in \mathcal{S}$, $\mathcal{G}_S \neq \emptyset$ and yet for any $\mathbf{U}$ that is independent of $(\mathbf{W}, \mathcal{G})$ and r-spread with $r = (64 \log k)/\epsilon$, we have $\mathbb{E}[|\mathcal{G}_{\mathbf{U}}|] < 1/8$.

Let us first use the theorem to complete our proofs of the sunflower lemma and the threshold vs expectation threshold conjecture.

7.1 Sunflower Lemma

To prove Claim 8, let $\mathcal{S}$ be the given family, and let $\mathbf{U}$ be a uniformly random set of $\mathcal{S}$, which is r-spread with $r = (64 \log k)/\epsilon$. If $\mathbf{W}, \mathcal{G}$ are as in the theorem, then $\mathbb{E}[|\mathcal{G}_{\mathbf{U}}|] < 1/8$ implies that $\mathbb{P}[\forall S \in \mathcal{S}, \mathcal{G}_S \neq \emptyset] < 1/8$ and so $\mathbb{P}[\mathcal{S}_{\mathbf{W}} \neq \emptyset] > 7/8$, proving Claim 8.

7.2 Threshold vs Expectation Threshold

Let $\mathcal{S} = \mathcal{F}$ be the family of minimal sets of the given monotone function f and $k = n$. Let g denote the monotone function whose family of minimal sets is $\mathcal{G}$. Let ϵ be the threshold of f, so $\mathbb{E}[f(\mathbf{X})] = 1/2$. Standard concentration bounds imply that there must be a number t with $|t - \epsilon n| < o(n)$ such that if $\mathbf{W}$ is chosen to be a uniformly random set with $|\mathbf{W}| = t$, then $\mathbb{E}[f(\mathbf{W})] \leq 3/4$. For ease of presentation, let us assume that $t = \epsilon n$. The first condition of the theorem asserts that either $f(\mathbf{W}) = 1$ or $f \leq g$, so

$$\mathbb{P}[f \leq g] \geq 1/4. \tag{7.1}$$

If $\mathbf{U}$ is the distribution on sets where each element is included in $\mathbf{U}$ independently with probability $1/r = \epsilon/(64 \log k)$, then $\mathbf{U}$ is r-spread, so the theorem guarantees that $\mathbb{E}[|\mathcal{G}_{\mathbf{U}}|] < 1/8$. By Markov's inequality:

$$\mathop{\mathbb{P}}_{\mathcal{G}}\left[\mathop{\mathbb{E}}_{\mathbf{U}}[|\mathcal{G}_{\mathbf{U}}|] \geq 1/2\right] < 1/4. \tag{7.2}$$

But (7.1) and (7.2) imply that there is a fixed choice of $\mathcal{G}$ such that $f \leq g$ and $\mathbb{E}[|\mathcal{G}_{\mathbf{U}}|] \leq 1/2$, proving the threshold vs expectation threshold conjecture.

7.3 Proving Theorem 9

Let $\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_{\log k}$ be uniformly random disjoint sets of size $m = \epsilon n / \log k$. Here all logarithms are computed base 2. Our goal is to use $\mathbf{W}_1, \ldots, \mathbf{W}_{\log k}$ to define sets $\mathcal{G}_1, \ldots, \mathcal{G}_{\log k}$. We shall then set $\mathbf{W} = \mathbf{W}_1 \cup \ldots \cup \mathbf{W}_{\log k}$ and $\mathcal{G} = \mathcal{G}_1 \cup \ldots \cup \mathcal{G}_{\log k}$.

Let $\mathbf{W}^i$ denote $\mathbf{W}_1 \cup \ldots \cup \mathbf{W}_i$, and $\mathcal{G}^i$ denote $\mathcal{G}_1 \cup \ldots \cup \mathcal{G}_i$. Define $\mathcal{G}_1, \ldots, \mathcal{G}_{\log k}$ iteratively as follows. For each i, and $S \in \mathcal{S}$, include $S - \mathbf{W}^i$ in $\mathcal{G}_i$ if and only if

(i). $|S - \mathbf{W}^i| \geq k/2^i$, and

(ii). $S - \mathbf{W}^i$ is a minimal set of $\{S - \mathbf{W}^i : S \in \mathcal{S}, \mathcal{G}_S^{i-1} = \emptyset\}$.

Intuitively, the above process attempts to *cover* all the sets of $\mathcal{S}$. In each step, we discard the elements of $\mathcal{S}$ that have already been covered, and proceed to cover more elements by including sets of size at least $k/2^i$ in $\mathcal{G}_i$. By the time $i = \log k$, we will cover all remaining sets that are not included in $\mathbf{W}_1 \cup \ldots \cup \mathbf{W}_{\log k}$. So, a set of $\mathcal{S}$ is left uncovered in this process only if it is contained in $\mathbf{W} = \mathbf{W}^{\log k}$.

We give an upper bound on the expected number of sets $T \in \mathcal{G}_i$ of size a as follows. Fix $\mathbf{W}_1, \ldots, \mathbf{W}_{i-1}$.

(i). There are at most $\binom{n}{m+a}$ choices for the set $T \cup \mathbf{W}_i$ with $|T| = a$. We have

$$\binom{n}{m+a} = \binom{n}{m} \cdot \prod_{j=1}^{a} \frac{n-m-j}{m+j} \leq \binom{n}{m} \cdot \left(\frac{n}{m}\right)^a = \binom{n}{m} \cdot \left(\frac{\log k}{\epsilon}\right)^a.$$

(ii). Given $T \cup \mathbf{W}_i$, let $S' - \mathbf{W}^{i-1}$ be the smallest set of $\{S - \mathbf{W}^{i-1} : S \in \mathcal{S}, \mathcal{G}_S^{i-1}\}$ that is contained in $T \cup \mathbf{W}_i$; break ties by picking the lexicographically first set. It must be that $|S' - \mathbf{W}^{i-1}| \leq k/2^{i-1}$, or else $S' - \mathbf{W}^{i-1}$ would have been included in $\mathcal{G}_{i-1}$. Furthermore, $T \subseteq S' - \mathbf{W}^{i-1}$, or $S' - \mathbf{W}^i$ would be a strict subset of T, and T would not be included in $\mathcal{G}_i$. So, there can be at most $2^{k/2^{i-1}} = 4^{k/2^i}$ choices for T consistent with $T \cup \mathbf{W}_i$.

The above count shows that the expected number of sets of size a in $\mathcal{G}_i$ is at most $4^{k/2^i}\left(\frac{\log k}{\epsilon}\right)^a$. Thus, we can bound

$$\begin{aligned}
\mathbb{E}[|\mathcal{G}_{\mathbf{U}}|] &\leq \mathbb{E}\Big[\sum_{Y \in \mathcal{G}} \Big(\frac{\epsilon}{64 \log k}\Big)^{|Y|}\Big] \\
&= \sum_{i=1}^{\log k} \mathbb{E}\Big[\sum_{Y \in \mathcal{G}_i} \Big(\frac{\epsilon}{64 \log k}\Big)^{|Y|}\Big] \\
&\leq \sum_{i=1}^{\log k} \sum_{a=k/2^i}^{\infty} \Big(\frac{\epsilon}{64 \log k}\Big)^a \cdot 4^{k/2^i} \Big(\frac{\log k}{\epsilon}\Big)^a \\
&= \sum_{i=1}^{\log k} \frac{(1/16)^{k/2^i}}{1 - 1/64} \\
&< \sum_{j=1}^{\infty} 2 \cdot (1/16)^j < 1/8.
\end{aligned}$$

This proves the theorem.

8 Conclusion

Sunflowers have had an enormous impact in a surprising number of different fields. They are certain to spring up in new places in the future. The counting method of [2] has already found applications in places where there are no sunflowers. It is an exciting time to be playing with these concepts!

9 Acknowledgements

Thanks to Paul Beame for useful comments.

References

[1] Miklos Ajtai and Endre Szemeredi. *Stud. Sci. Math. Hungar*, (9):9–11, 1974.

[2] Ryan Alweiss, Shachar Lovett, Kewen Wu, and Jiapeng Zhang. Improved bounds for the sunflower lemma. In *Proccedings of the 52nd Annual Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 624–630. ACM, 2020.

[3] Tolson Bell, Suchakree Chueluecha, and Lutz Warnke. Note on sunflowers. *Discret. Math.*, 344(7):112367, 2021.

[4] Paul Erdős and Richard Rado. Intersection theorems for systems of sets. *Journal of London Mathematical Society*, 35:85–90, 1960.

[5] Paul Erdös and András Sárközy. Arithmetic progressions in subset sums. *Discret. Math.*, 102(3):249–264, 1992.

[6] Gudmund Skovbjerg Frandsen, Peter Bro Miltersen, and Sven Skyum. Dynamic word problems. *J. ACM*, 44(2):257–271, 1997.

[7] Keith Frankston, Jeff Kahn, Bhargav Narayanan, and Jinyoung Park. Thresholds versus fractional expectation-thresholds. *CoRR*, abs/1910.13433, 2019.

[8] Anna Gál and Peter Bro Miltersen. The cell probe complexity of succinct data structures. *Theor. Comput. Sci*, 379(3):405–417, 2007.

[9] A. W. Hales and R. I. Jewett. Regularity and positional games. *Transactions of the American Mathematical Society*, 106(2):222–229, 1963.

[10] Jeff Kahn and Gil Kalai. Thresholds and expectation thresholds. *Comb. Probab. Comput.*, 16(3):495–502, 2007.

[11] Jinyoung Park and Huy Tuan Pham. A proof of the kahn-kalai conjecture. *arXiv preprint arXiv:2203.17207*, 2022.

[12] Sivaramakrishnan Natarajan Ramamoorthy and Anup Rao. Lower bounds on non-adaptive data structures maintaining sets of numbers, from sunflowers. In *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*, pages 27:1–27:16, 2018.

[13] Anup Rao. Coding for sunflowers. *Discrete Analysis*, 2020.

[14] A. A. Razborov. Some lower bounds for the monotone complexity of some boolean functions. *Soviet Math. Dokl.*, 31(31):354–357, 1985.

[15] K. F. Roth. On Certain Sets of Integers. *Journal of the London Mathematical Society*, s1-28(1):104–109, 01 1953.

[16] Michel Talagrand. Are many small sets explicitly small? In Leonard J. Schulman, editor, *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 13–36. ACM, 2010.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il