In Search of Degree-4 Sum-of-Squares Lower Bounds for MaxCut

Corwin de Boor

CMU-CS-19-118
August 2019

Computer Science Department
School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213

Thesis Committee
Ryan O'Donnell, Chair
Venkatesan Guruswami

Submitted in partial fulfillment of the requirements for the Degree of Master of Science

Keywords: lower bounds, constraint satisfaction problems, average-case hardness, semidefinite programming, sum-of-squares, pseudo-expectation, d-regular random graphs, trace method, max-cut, high girth, d-regular infinite tree, pseudo-moments, poly-symmetry, positive semi-definite, configuration-symmetry, gaussian wave

Abstract

The behavior of MaxCut on large, regular, random undirected graphs is well understood up to third-order terms. Even so, we do not yet know of efficient algorithms for refuting the existence of a large cut. Our current best-known refutation algorithms are all specific instances of the sum-of-squares (SOS) algorithms hierarchy, and we know degree-2 SOS does not well approximate the MaxCut on these random graphs. As the hierarchy gets more powerful with increasing degree, does this result hold for degree-4 SOS as well?

In this work, we develop a technique for extending degree-2 pseudo-expectations, objects showing inapproximability, to degree-4 pseudo-expectations. We identify configuration-symmetry as a key property of these pseudo-expectations, and bound the effects that prevent natural extension. By analyzing those effects more carefully, one may be able to obtain degree-4 SOS lower-bounds.

This thesis contains joint work with Ryan O'Donnell and Tselil Schramm.

Acknowledgements

I would like to thank my advisor Ryan O'Donnell for joining me on this wonderful adventure in matrix-land. The adventure sometimes continued into the wee hours of the morning or sometimes blazed throughout the entire day as we chased down pesky technicalities and suffered repeated jaunts back to square one. Though most of this work was done while not even on the same coast of the US, Ryan still managed to appear each Thursday to discuss the strategy of the week.

I would also like to thank Tselil Schramm, whose knowledge of configurations, shapes, the trace method, and all things sum-of-squares were unparalleled. Thank you for joining us in inching our way around the circle of psuedo-expectation (increasing the degree?).

Thank you to Venkat Guruswami, who agreed to be on my thesis committee even though I asked him well after than I should have.

Thank you also to Dave Eckhardt and Anil Ada, without whom I likely would not even have started this thesis. They both provided me much needed clarity in navigating the wonderful world of choices.

Of course, thank you also to Tracy Farbacher for making sure that I get through this program, and always having the answers to my questions.

Thank you to all my friends for their support and fun times during this thesis. Xinyu Wu, thank you for always asking about and listening to my current progress on the project. My roommates Matthew Savage and Zach Wade were sources of sanity throughout this thesis. I don't know what I would have done without Friday game night and Scythe-on-demand. The apartment is woeful without you guys, so thank you.

Thank you to my family, Mom, Dad, Jasper, and Malva, for putting up with the high-stress times near the end of this thesis. I really appreciated our 3-hour chat about my thesis because it solidified what background information I needed to include.

Monique Mezher, thank you for always being there for me when times were tough. Now that it's in a technical report, you know it's confirmed: wean.

Contents

1 Introduction

In the quest to understand whether a problem can be solved computationally, we often turn to worst-case hardness. There, NP-hardness tells us that a particular problem is at least as hard as any problem in NP—problems whose answers can be checked in polynomial time. Most natural problems that people want to solve can be expressed in NP—solving puzzles, determining whether a circuit has a valid input, graph coloring, finding dense clusters—so understanding this hardness has many applications. However, worst-case hardness does not give a complete picture of the difficulty of a problem. A problem that is known to be NP-hard could only have one super-difficult instance. In this case, we may be able to write an algorithm that gives answers quickly on almost all possible inputs and only fails to work on that one situation. For practical applications, this algorithm would be almost as good as an algorithm that works all the time; the practical instances that come up every day would be easily solved and the probability of getting the one bad instance would be negligible. How do we know whether a particular problem is going to be infeasible to solve in most cases? For that, we turn to average-case hardness: whether a random instance is going to be difficult to solve.

For determining whether a problem is hard in the average case, we turn to the sum-of-squares (SOS) algorithm family. The SOS algorithm takes a degree (strength/round) parameter and gets better, but slower, as the degree increases. Typically, SOS is applied on the refutation side of a problem: producing a checkable proof that the particular instance has no solution—no coloring, no valid input, no dense cluster. Our goal is to show that, unless the degree is sufficiently high, SOS will not be able to refute a random instance of our chosen problem. Of course, showing that a particular algorithm fails is not sufficient to argue that a problem is not solvable; however, understanding the behavior of SOS does give us a lot of information about the difficulty of the problem itself. In particular, SOS is the optimal approximation algorithm for many kinds of combinatorial optimization if the Unique-Games conjecture holds [21]. Also, many of the current best-known approximation algorithms for combinatorial optimization problems make use of spectral or semidefinite-programming methods [11]. As SOS is a particular generalization of semidefinite programming, understanding the behavior of SOS will tell us whether these methods can be extended.

The particular combinatorial optimization problem we will focus on is MaxCut. The MaxCut problem tries to isolate one particular aspect of graph connectivity: given an unweighted, undirected graph, we would like to separate its vertices into two parts such that the number of edges between the parts is as large as possible. Alternatively, we can think about painting each vertex either red or blue while maximizing the number of edges that touch both colors. As MaxCut is NP-hard, its worst-case behavior is well-understood. Recently, Montanari [18] came up with an optimal search algorithm in the average-case under the mild assumption of a conjecture from statistical physics.

But for average-case refutation, we understand MaxCut for degree-2 SOS [19, 4, 3, 11]. The dual object to an SOS proof is called a pseudo-expectation, and, as the dual to a refutation, it acts like a MaxCut partition. The main metric of a pseudo-expectation is its average edge-correlation ϵ. We know the true value of the MaxCut has edge-correlation $\epsilon \approx 2P_*/\sqrt{d} \approx 1.526/\sqrt{d}$ [10], which is the number to beat for any lower-bound. For degree-2 SOS, we have optimal pseudo-expectations with $\epsilon \approx 2/\sqrt{d}$. Our main goal is to extend this result to higher SOS degrees.

A week before the defense of this work, Kunisky and Bandeira [14] accomplished a very similar goal by creating a degree-4 psuedo-expectation with the same $\epsilon \approx 2/\sqrt{d}$ for the Sherrington-Kirkpatrick hamiltonian, rather than MaxCut. This work does not discuss that result.

1.1 Our Results

Our main result is a technique for extending a degree-2 pseudo-expectation into a degree-4 pseudo-expectation. To do this, we isolate a key property of all the known degree-2 pseudo-expectations that arises from our graph distribution. We generalize this property to a notion called configuration-symmetry. With configuration-symmetry and a degree-2 pseudo-expectation, many decisions for how to construct the degree-4 pseudo-expectation are forced. In our analysis of the spectrum of this degree-4 pseudo-expectation, we were only able to achieve achieve $\epsilon \approx 1/(6\sqrt{d})$. We believe our construction does present a non-trivial degree-4 pseudo-expectation because it is motivated by the structure of the problem, even though we were not able to fully analyze it. In service to this main result, we prove a lemma that converts, with a negligible loss, pseudo-expectations on regular graphs of high girth to pseudo-expectations on random regular graphs.

2 Polynomial Proofs

2.1 MaxCut as a Polynomial

Let $G = (V, E)$ be a simple, undirected graph and A_G be its 0/1-indicator adjacency matrix. If $u, v \in V$ are vertices in G then the entry $(A_G)_{u,v}$ is 1 if $\{u, v\} \in E$ is an edge of G and 0 otherwise. We will often write $u, v \in G$ to mean vertices u and v are vertices of G and $e \in G$ if the edge $e = uv = \{u, v\}$ is an edge of G. If the graph G is unambiguous, we will write $A = A_G$ for the adjacency matrix.

MaxCut is most naturally stated its optimization form:

Problem. Given a simple, undirected graph $G = (V, E)$, find the set $S \subseteq V$ maximizing the number of crossing (cut) edges $|\{uv \in E : u \in S, v \notin S\}|$.

We can rewrite this problem into a slightly different form. Suppose we have some variables $x_v \in \{\pm 1\}$ for each vertex $v \in G$. We could identity $x_v = 1$ with $v \in S$ and $x_v = -1$ with $v \notin S$. For an edge $uv \in G$, the product $x_u x_v = -1$ if and only if exactly one of u and v is in S and the edge uv is cut. We can rewrite the objective

$$|\{uv \in E : u \in S, v \notin S\}| = \sum_{uv \in E} \frac{1 - x_u x_v}{2}$$

and restate MaxCut as

Problem. Given a simple, undirected graph G, maximize $\sum_{uv \in E} -x_u x_v$. where $x_v \in \{\pm 1\}$.

2.2 The Sum-of-Squares Algorithm

We can also consider the refutation variant of MaxCut:

Problem. Given a simple, undirected graph $G = (V, E)$ of size $|V| = n$ and constant k, output a "checkable proof" that every choice of $x \in \{\pm 1\}^n$ satisfies $k - \sum_{uv \in E} -x_u x_v \geq 0$ if such a proof exists.

By using standard search-to-decision techniques, an algorithm for the refutation variant of this problem can efficiently be transformed into an algorithm for the optimization variant.

But, how would we actually give such a proof? We are trying to show that our target polynomial $f(x) = k - \sum_{uv \in E} -x_u x_v$ is non-negative for every $x \in \{\pm 1\}^n$. Of course, the square of a polynomial must be non-negative everywhere, so if we could find a set of polynomials $g_1, \dots, g_t$ such that

$$f(x) = g_1(x)^2 + \cdots + g_t(x)^2$$

for all choices of $x \in \{\pm 1\}^n$ then this would show that $f \geq 0$. We call the polynomials $g_1, \dots, g_t$ a degree-$2r$ sum-of-squares certificate of the fact $f \geq 0$ if the polynomials $g_1, \dots, g_t$ have degree at most r.

As of yet, this certificate does not seem very useful; we would still need to verify the equality for all $x \in \{\pm 1\}^n$. To fix this, we will multilinearize the polynomial. Say a polynomial $p(x)$ is multilinear if it can be written as

$$p(x) = \sum_{S \subseteq [n]} \left(\alpha_S \prod_{v \in S} x_v \right)$$

for some coefficients $\alpha_S \in \mathbb{R}$. Because every variable $x_v \in \{\pm 1\}$, its square $x_v^2 = 1$. By repeatedly replacing $x_v^c \mapsto x_v^{c \bmod 2}$ within a polynomial p, we multilinearize p into the polynomial $\hat{p}$ equal to p on $\{\pm 1\}^n$. This multilinear representation is unique [20], so the certificate is good if the multilinearization of $f - g_1^2 - \cdots - g_t^2$ is identically zero.

With an actual structure of our proofs in hand, we can go about looking for them. As it turns out, a generalization of the Positivstellensatz tells us that

Fact 2.1 (Corrollary of Positivstellensatz)**.** *For $f : \{\pm 1\}^n \to \mathbb{R}$ a polynomial, there exists degree-$2n$ sum-of-squares certificates of the fact $f \geq 0$.*

Proof. Explicitly construct the degree-n multilinear polynomial $g(x) = \sqrt{f(x)}$ that has one term for each $x \in \{\pm 1\}^n$. For example, when $n = 1$ construct the function

$$g(x) = \frac{\sqrt{f(1)} + \sqrt{f(-1)}}{2} \cdot 1 + \frac{\sqrt{f(1)} - \sqrt{f(-1)}}{2} \cdot x$$

By construction, we have that $g^2 = f$ on $x \in \{\pm 1\}^n$. As g had degree n, it serves as a degree-$2n$ sum-of-squares certificate. □

The sum-of-squares algorithm hierarchy looks for degree-r sum-of-squares certificates that $f \geq 0$ for a particular r. To do this, it constructs a semidefinite program optimizing over $n^{O(r)}$ variables. Thus, for r being a constant, this algorithm runs in polynomial time.

2.3 Pseudo-expectations

The main concern of this work is the dual of the previous question: when do there *not* exist degree-r SOS certificates that the polynomial $f \geq 0$? Thinking all the way back to where these certificates originated, the SOS certificate was just a way to tell us that the objective $f(x) \geq 0$ for all choices of x. One way that we could show no certificate exists would be by exhibiting some particular value of x violating the objective with $f(x) < 0$. We could even choose a distribution D over the x values such that

$$\mathop{\mathbb{E}}_{x \sim D}[f(x)] < 0$$

But choosing such a distribution will not let us prove anything useful. We already argued that the SOS certificate only can exist if f is truly non-negative, so, for MaxCut, there will not be SOS certificates of any degree that show us the cut must be smaller than the true MaxCut.

For our purposes we want to be able to argue something stronger: if r is too small, the degree-r SOS certificates will not be able to get anything close to the true MaxCut. To do this, we will create a fake distribution on the x to fool all of the degree-r certificates. We call the linear functional[1] $\widetilde{\mathbb{E}}_x : \mathbb{R}[x] \to \mathbb{R}$ a degree-r pseudo-expectation if it satifies

- scaling: $\widetilde{\mathbb{E}}_x[1] = 1$,
- positivity: $\widetilde{\mathbb{E}}_x[p(x)^2] \geq 0$ for every polynomial p^2 of degree at most r, and
- poly-symmetry: $\widetilde{\mathbb{E}}_x[p(x)] = \widetilde{\mathbb{E}}_x[\hat{p}(x)]$ for every polynomial p and its corresponding multilinearization $\hat{p}$.

The most important consideration when choosing our pseudo-expectation is that it helps: our target polynomial must satisfy $\widetilde{\mathbb{E}}[f(x)] < 0$.

Does this pseudo-expectation actually accomplish our intended goal? Suppose we have a degree-r pseudo-expectation with $\widetilde{\mathbb{E}}[f(x)] < 0$ and a purported degree-r SOS certificate $g_1, \ldots, g_t$. Then, we can derive a contradiction

$$0 = \widetilde{\mathbb{E}}[0] = \widetilde{\mathbb{E}}[f(x) - g_1(x)^2 - \cdots - g_t(x)^2] = \widetilde{\mathbb{E}}[f(x)] - \widetilde{\mathbb{E}}[g_1(x)^2] - \cdots - \widetilde{\mathbb{E}}[g_t(x)^2] < 0$$

where we used poly-symmetry, linearity, and then positivity. Thus, the existence of a degree-r pseudo-expectation actually does prevent a degree-r SOS certificate. The rest of this work will focus on creating psuedo-expectations to show the failure of the SOS algorithm.

For more information and an exhaustive treatment of the sum-of-squares algorithm, SOS certificates, pseudo-expectations, and applications, see the survey paper by Barak and Steurer [5] or the paper on planted clique by Barak et al. [6]. There, scaling and poly-symmetry are known as "hard" constraints because they must be satisfied exactly, and positivity is known as a "soft" constraint.

[1] A linear operator that maps polynomials to real numbers

2.4 The Moment Matrix

How exactly do we define a degree-$2r$ psuedo-expectation $\widetilde{\mathbb{E}}$? Because $\widetilde{\mathbb{E}}$ satisfies poly-symmetry, it is sufficient (and necessary by uniqueness of the multilinear representation) to define $\widetilde{\mathbb{E}}$ only on multilinear polynomials; the rest are filled in by looking at their multilinear counterpart. Then, by linearity it suffices (and is necessary) to define $\widetilde{\mathbb{E}}$ on monomials: $\{\prod_{v\in S} x_v : S \subseteq [n]\}$. Finally, we need to somehow ensure positivity, that $\widetilde{\mathbb{E}}[p(x)^2] \geq 0$ for all $p(x)^2$ of degree at most $2r$. Because positivity only depends on monomials of degree at most $2r$, we can map monomials of higher degree arbitrarily.

To ensure positivity, let us examine the vector $\mathrm{mon}_r(x)$ indexed by subsets $S \subseteq [n]$ of size $|S| \leq r$, where the entry $(\mathrm{mon}_r(x))_S = \prod_{i\in S} x_i$. Let m be the dimension of $\mathrm{mon}_r(x)$. Define the moment matrix

$$M = \widetilde{\mathbb{E}}[\mathrm{mon}_r(x) \cdot \mathrm{mon}_r(x)^\intercal]$$

whose entries contain only and all of the terms $\widetilde{\mathbb{E}}[p(x)]$ where $p(x)$ is a monomial of degree at most $2r$.

Lemma 2.1. *$\widetilde{\mathbb{E}}$ satisfies positivity if and only if the matrix $M = \widetilde{\mathbb{E}}[\mathrm{mon}_r(x) \cdot \mathrm{mon}_r(x)^\intercal] \succeq 0$.*

Proof. In the forward direction, take an arbitrary vector $v \in \mathbb{R}^m$ and compute the quadratic form

$$v^\intercal M v = \widetilde{\mathbb{E}}[v^\intercal \mathrm{mon}_r(x) \cdot \mathrm{mon}_r(x)^\intercal v] = \widetilde{\mathbb{E}}[v^\intercal \mathrm{mon}_r(x) \cdot \mathrm{mon}_r(x)^\intercal v] = \widetilde{\mathbb{E}}[(\mathrm{mon}_r(x)^\intercal v)^2] \geq 0$$

In the reverse direction, let $p(x)$ be some polynomial of degree at most r. We can write

$$\hat{p}(x) = \sum_{S\subseteq[n]:|S|\leq r} \left(v_S \cdot \prod_{i\in S} x_i \right) = v^\intercal \mathrm{mon}_r(x)$$

for a vector $v \in \mathbb{R}^m$. Then, the pseudo-expectation

$$\widetilde{\mathbb{E}}[p(x)^2] = \widetilde{\mathbb{E}}[(v^\intercal \mathrm{mon}_r(x))^2] = \widetilde{\mathbb{E}}[v^\intercal \mathrm{mon}_r(x) \cdot \mathrm{mon}_r(x)^\intercal v] = v^\intercal M v \geq 0$$

□

In addition to positivity, we need to define poly-symmetry for the moment matrix M. Partition the entries indexed by (S, T) of M into equivalence classes based on the multi-linearization of $\prod_{i\in S} x_i \cdot \prod_{i\in T} x_i$ (equivalently by $S \triangle T$). As the values in each equivalence class correspond to the same multilinear polynomial, our pseudo-expectation will give the same value by poly-symmetry. Conversely, any two cells in different equivalence classes have different polynomials, so we have free choice as to what their values are. Finally, our scaling property is $M_{\emptyset,\emptyset} = 1$ (which holds if and only if $\widetilde{\mathbb{E}}[1] = 1$). We will often bundle scaling in with poly-symmetry.

To pick the pseudo-expectation $\widetilde{\mathbb{E}}$, it suffices to pick a poly-symmetric matrix $M \succeq 0$ as its moment matrix. We will call such a moment matrix M **valid**.

2.5 The Trace Method

One way we might want to show positivity for the moment matrix M of dimension n is to show $M = 1 + A$ for a spectrally small matrix A. We define the spectral radius

$$\operatorname{spr}(A) = \max_{v:\|v\|=1} \|Av\|$$

Because A is symmetric, the spectral radius $\operatorname{spr}(A) \in \{|\lambda_1|, |\lambda_n|\}$ where the eigenvalues of A are $\lambda_1 \geq \cdots \geq \lambda_n$. Then, the quadratic form $v^\intercal M v = v^\intercal v + v^\intercal A v \geq v^\intercal v(1 - \operatorname{spr}(A))$. Instead of showing positivity, we can bound the spectral radius of A.

To isolate the spectral radius of A, we look at its long-term behavior after T applications. For our chosen even T, the eigenvalues of the power A^T are $\lambda_1^T, \ldots, \lambda_n^T$. This tells us the trace is bounded by

$$\operatorname{Tr}[A^T] = \sum_{i=1}^{n} \lambda_i^T \geq \operatorname{spr}(A)^T \qquad\qquad \operatorname{Tr}[A^T] = \sum_{i=1}^{n} \lambda_i^T \leq n \cdot \operatorname{spr}(A)^T$$

because $\operatorname{spr}(A) \in \{|\lambda_1|, |\lambda_n|\}$ and all of the eigenvalue powers satisfy $0 \leq \lambda_i^T \leq \operatorname{spr}(A)^T$ for even T. By choosing the power $T \geq C \lg n$, we can obtain a very close approximation of the spectral radius

$$\operatorname{spr}(A) \leq \sqrt[T]{\operatorname{Tr}[A^T]} \leq n^{1/T} \cdot \operatorname{spr}(A) \leq 2^{1/C} \operatorname{spr}(A)$$

If we think about A as the (weighted) adjacency matrix of a graph, computing the trace of a high power T counts the total weight of closed walks starting and ending at any vertex. Then, we can use our intuition about the graph in order to count these walks. The trace method is useful because it enables us to measure a spectral property in an combinatorial way. In this work, we will use the trace method to analyze "glitches" in our prepared degree-4 moments.

3 Large Regular Random Undirected Graphs

In order to analyze average-case complexity, we need to define our average-case—a distribution over graphs G. Given the degree d and the size n, we consider the uniform distribution $\mathcal{G}_{n,d}^{\text{reg}}$ over random d-regular graphs with n vertices. This model is discussed in detail by Wormald [22]. We would like to think about asymptotic behavior with n being much larger than d; namely, we will think of d as a constant with respect to n becoming large, but then also look at the asymptotic behavior with respect to d. For the purposes of imagining these graphs, think of $d = 17$ and $n = 10^6$.

One could also consider the more popular Erdös-Rényi distribution: $G \sim \mathcal{G}(n, d/n)$. There, each pair of vertices has an edge independently with probability d/n. However, this makes the spectrum of the adjacency matrix A_G much more difficult to analyze and the resulting graphs G much less nice. In particular, while graphs $G \sim \mathcal{G}_{n,d}^{\text{reg}}$ are connected with high probability [22], graphs $G \sim \mathcal{G}(n, d/n)$ have many disconnected vertices; each vertex has a constant probability (related to d) of being disconnected. As such, we will confine ourselves to the random regular distribution.

3.1 The Truth about MaxCut

As it turns out, the true value of the MaxCut is actually known for graphs $G \sim \mathcal{G}_{n,d}^{\text{reg}}$. By using an approach from statistical physics, Dembo et al. [10] were able to determine that for random regular graphs $G \sim \mathcal{G}_{n,d}^{\text{reg}}$ the cut value is, with high probability,

$$\text{MaxCut}(G) = n\left(\tfrac{d}{4} + P_*\sqrt{\tfrac{d}{4}} + o(\sqrt{d})\right) + o(n)$$

where $P_* \approx 0.7632$ is an analytic constant coming from a ground-state energy as defined in Dembo et al. [10]. Recall that in the situation where $x \in \{\pm 1\}^n$ is the optimal partition, we can express

$$\text{MaxCut}(G) = \sum_{uv \in E} \frac{1 - x_u x_v}{2} = \frac{nd}{4} + \sum_{uv \in E} \frac{-x_u x_v}{2}$$

Rewriting these two equations, we find that

$$2/(nd) \sum_{uv \in E} -x_u x_v = 2P_*/\sqrt{d} + o(1/\sqrt{d}) \approx 1.526/\sqrt{d}$$

where $nd/2 = |E|$. Define this quantity, the average cutting of each edge, as

$$\epsilon := 2/(nd) \sum_{uv \in E} -x_u x_v$$

We will seek to make ϵ as large as possible.

3.2 Being the Barber

You might ask whether there exist efficient algorithms for finding the partition x to make ϵ as large as possible. Remember that our goal for this work is to find psuedo-expectations that act like fake cuts in order to fool the sum-of-squares algorithm. As it turns out, we can use a fake cuts from a pseudo-expectation in order to generate a real partition x. This rounding technique is due to Goemans and Williamson [13]. The below algorithm will be very similar to what they originally described but presented from a different perspective.

Let us suppose we are given a graph G on n vertices and we have a procedure to construct, with high probability, a degree-2 pseudo-expectation $\widetilde{\mathbb{E}} = \widetilde{\mathbb{E}}_G$ where $-1 \leq \widetilde{\mathbb{E}}[x_i x_j] \leq 0$ for $ij \in E(G)$. We want this pseudo-expectation to be a good fake cut, namely

$$\widetilde{\mathbb{E}}\left[\textstyle\sum_{ij \in E} x_i x_j\right] = -\epsilon |E|$$

for $\epsilon \geq (2 - o_n(1))/\sqrt{d}$. We will describe several algorithms for obtaining a pseudo-expectation later on; all of the known results satisfy these requirements.

Now, let's look at the (partial) moment matrix $M = \widetilde{\mathbb{E}}[x \cdot x^\intercal] \succeq 0$. We know that we can find a set of vectors v_i such that $M_{ij} = v_i^\intercal v_j$ because $M \succeq 0$. Note that the scaling property of M implies that $\|v_i\| = 1$. To find our actual partition (x_i), we will randomly round the vectors (v_i). For this, we will pick a random direction r and set $x_i = \text{sgn}(r^\intercal v_i)$. Picking this random directly r picks a random hyperplane and we round the vectors according to which side of the hyperplane they appear on.

We will cut an edge ij ($x_i x_j = -1$) if the two vectors v_i and v_j appear on opposite sides of the hyperplane. This happens with probability proportional to the angle between them. As such, the probability $\Pr[x_i x_j = -1] = \frac{1}{\pi}\arccos(v_i^\intercal v_j)$ [13]. Using the fact that $1 - \frac{2}{\pi}\arccos(y) \leq \frac{2}{\pi}y$ for $-1 \leq y \leq 0$, we can compute the expected value gotten by this rounding scheme:

$$\mathbb{E}\left[\sum_{ij\in E} x_i x_j\right] = \sum_{ij\in E}(1 - \tfrac{2}{\pi}\arccos(v_i^\intercal v_j)) \leq \sum_{ij\in E}\frac{2}{\pi}v_i^\intercal v_j = \frac{2}{\pi}\widetilde{\mathbb{E}}\left[\sum_{ij\in E} x_i x_j\right]$$

which gives us an edge coefficient of

$$\epsilon \geq (\tfrac{4}{\pi} - o_n(1))/\sqrt{d} \approx 1.273/\sqrt{d}$$

Recently, Montanari [18] came out with a better algorithm for the $\mathcal{G}(n, d/n)$ model. The correctness proof also relies on an unproven conjecture from statistical physics called continuous replica symmetry breaking; the conjecture is widely believed to hold with evidence supporting it. The algorithm from Montanari [18] makes use of a technique they call incremental approximate message passing, which iteratively refines its solution towards a good objective value by mixing across edges. With high probability over the choice of graph, they achieve an edge correlation of

$$\epsilon = (1 - \delta)2P_*/\sqrt{d} - o(1/\sqrt{d})$$

for any $\delta > 0$, which is effectively optimal. The algorithm runs in polynomial time if you treat δ as a constant.

3.3 A Tree Near You

One nice property of graphs $G \sim \mathcal{G}_{n,d}^{\text{reg}}$ is that they are locally tree-like. Most vertices in G have neighborhoods that look like trees. Indeed, let L be a constant. The L-neighborhood of a vertex v is the set

$$N_L(v) = \{u \in G : d(u, v) \leq L\}$$

If the vertex v has a cycle in its neighborhood $N_L(v)$ then we call it L-bad. With high probability, the number of L-bad vertices in G is at most $\log n$ [22].

Because so few vertices do not have tree-like neighborhoods, we will now only consider graphs G with girth $g(G) \geq \Omega(L)$. Then, every vertex will have a tree-like neighborhood. After finding results for high-girth graphs, we will be able to translate the results down to all graphs with a negligible loss.

3.4 Configuration Symme-tree

Combining regularity and high girth, the neighborhoods of every vertex looks the same. This motivates a strengthened requirement on $\widetilde{\mathbb{E}}$: configuration symmetry. Configuration-symmetry is a generalization of invariants of the prior work's degree-2 psuedo-expectations.

Let $U \in V(G)^r$ be an ordered r-tuple of vertices (duplicates allowed). Define the L-configuration $C_L(U) = (H, U)$ as follows: H is the minimal subgraph of G containing all vertices in U and, for each $a, b \in U$ where $d(a, b) \leq L$, all vertices on the shortest path $P_{a,b}$ between a and b. See Figure 1 for an example configuration. By requiring the girth $g(G) > rL$, the shortest path $P_{a,b}$ is unique (if it exists) and $H(C_L(U))$ will be a forest.

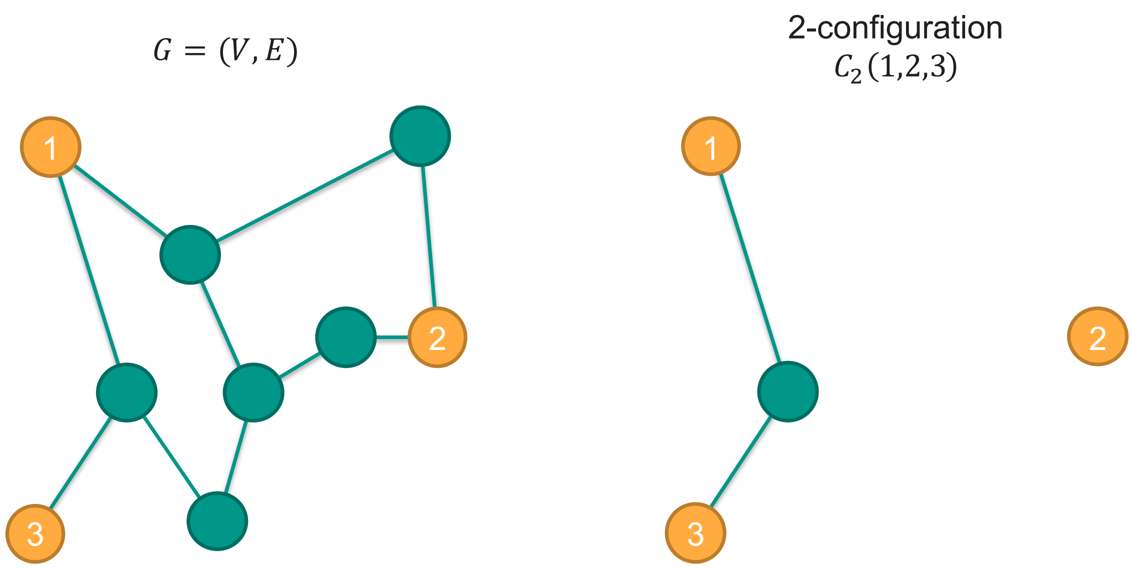

Figure 1: An example configuration with $L = 2$

We say that two configurations C and D are isomorphic $C \simeq D$ if there exists some vertex bijection $f : V(C) \to V(D)$ such that relabeling the vertices in C results in D. In other words, there is an isomorphism between their underlying graphs $H(C)$ and $H(D)$ that preserves the identified vertices $U(C)$ and $U(D)$. We will naturally extend other graph operations to configurations by having them work directly on the graph H.

We can now talk about what it means for the pseudo-expectation $\widetilde{\mathbb{E}}$ to be L-configuration-symmetric. Given any two L-configurations C and D, the pseudo-expectation

$$\widetilde{\mathbb{E}}\left[\prod_{i \in U(C)} x_i\right] = \widetilde{\mathbb{E}}\left[\prod_{i \in U(D)} x_i\right]$$

We can extend this notion to an equivalent one on the moment matrix M just as we did with poly-symmetry. For M to be configuration-symmetric, we require that two entries with the same configuration have the same value. The L-configuration of an entry (S, T) is $C_L(S, T)$, where we convert a set into a tuple by sorting its elements.

3.5 Perfect Trees

One way to ensure configuration-symmetry by design is to study, instead of the finite graphs G, the d-regular infinite tree T_d. The infinite tree T_d is useful because it *actually has* certain desirable properties our random regular graphs G almost have. For example, the adjacency operator A_{T_d} has spectral radius $\text{spr}(A_{T_d}) \leq 2\sqrt{d-1}$ [12]. This spectral radius is exactly what we would expect from a random regular graph if not for the troublesome all-ones eigenvector.

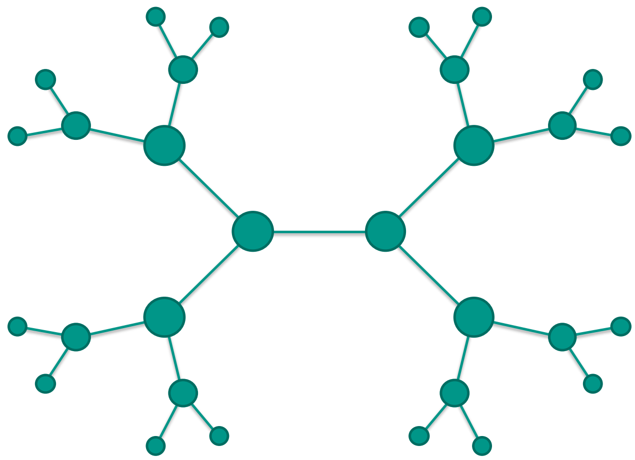

Figure 2: Part of the 3-regular infinite tree T_3

Because every vertex in the infinite tree has an automorphism to every other vertex, it really only makes sense to consider psuedo-expectations that are automorphism-invariant. This automorphism-invariance is what provides configuration-symmetry.

4 Degree-2 Pseudo-expectations

Because we will use degree-2 pseudo-expectations as the basis for our degree-4 pseudo-expectations, we will take some time to review the degree-2 constructions. Although the degree-2 constructions all satisfy configuration-symmetry and are very similar, we will show that there is some leeway in the choices of the moments. This leeway is unexpected because configuration-symmetry initially seems restrictive.

The constructions we will discuss all define the moment matrix M restricted to singleton rows and columns (indexed by S with $|S| = 1$) rather than defining the whole moment matrix. We can equivalently consider M indexed by vertices.

Lemma 4.1. *If a matrix M with rows and columns indexed by the vertices V is positive semi-definite $M \succeq 0$ and has 1 on the diagonal ($M_{i,i} = 1$), then the full moment matrix*

$$M' = \widetilde{\mathbb{E}}[\mathrm{mon}_1(x) \cdot \mathrm{mon}_1(x)^\intercal] = \begin{array}{c} {\scriptstyle S,T} \\ {\scriptstyle |S|=0} \\ {\scriptstyle |S|=1} \end{array} \begin{array}{c} \begin{array}{cc} {\scriptstyle |T|=0} & {\scriptstyle |T|=1} \end{array} \\ \left[\begin{array}{cc} 1 & 0 \\ 0 & M \end{array}\right] \end{array}$$

is valid.

Proof. Clearly, the matrix $M' \succeq 0$ because it is PSD block-diagonal. Scaling is satisfied because $M'_{\emptyset,\emptyset} = 1$. Therefore, we only need to check that M' is poly-symmetric. We do this by casing on the size of $S \triangle T$ for entry (S, T) of M'.

- $|S \triangle T| = 0$. Then, $S = T$ and we are considering a diagonal entry. By assumption $M_{S,S} = 1$ and by construction $M'_{\emptyset,\emptyset} = 1$, so these entries all have the same value.

- $|S \triangle T| = 1$. Then, $S = \emptyset$ and $|T| = 1$ or vice-versa. In either case, $M'_{S,T} = 0$ by construction.

- $|S \triangle T| = 2$. Then, we know $S \neq T$ and $|S| = |T| = 1$, so we lie squarely within M. In order to obtain the same $S \triangle T$, our sets must either be (S, T) or (T, S). By symmetry of M, these have the same value. □

Why did we enforce that $\widetilde{\mathbb{E}}[x_i] = 0$? A pseudo-expectation pretends to be a coloring of the graph with a large cut. For any particular coloring, flipping the colors of every vertex does not change the cut. Thus, for each coloring where a vertex i is colored $+1$, there is an equivalent coloring where vertex i is colored -1. Intuitively, we lose nothing by enforcing that $\widetilde{\mathbb{E}}[x_i] = 0$.

Our final perspective on the pseudo-expectation comes from the following fact

Fact 4.1. *A matrix $M \succeq 0$ if and only if there exist centered Gaussian random variables X_i with moment matrix $\mathbb{E}[XX^\intercal] = M$.*

Proof. Take a vector of i.i.d standard Gaussians Y and mix them to form $X = VY$ where $VV^\intercal = M$. The X are still centered and

$$\mathbb{E}[XX^\intercal] = V\,\mathbb{E}[YY^\intercal]\,V^\intercal = V I\,V^\intercal = M$$ □

Thus, it will be sufficient choosing a degree-2 pseudo-expectation to pick a standard Gaussian for each vertex. Recall that our goal is to have a pseudo-expectation where

$$\widetilde{\mathbb{E}}\left[k - \textstyle\sum_{uv \in E} -x_u x_v\right] < 0$$

to show that we cannot refute the existence of the MaxCut. As such, we need to pick Gaussians with large negative correlations across edges in the graph. Once we have chosen these Gaussians, we can assign the matrix M to be the moment matrix of the Gaussians, further extend it to the full moment matrix M', and thereby define the pseudo-expectation.

4.1 To Infinitree and Beyond

The first degree-2 psuedo-expectation comes from thinking about the Gaussian perspective on the d-regular infinite tree T_d [10]. That is, we will try to pick a standard Gaussian for each vertex of the tree T_d. The Gaussians chosen will have some good edge correlations and also the property that two far-enough vertices have zero correlation. This far-enough property is what enables us to then move the Gaussians down to a finite graph.

Indeed, the far-enough property is equivalent to L-configuration-symmetry. By focusing on the infinite tree and restricting ourselves to automorphism-invariant psuedo-expectations, we get that any two paths of the same length have the same pseudo-expectation. But if we look at L-configurations instead of ∞-configurations, we only have paths up to length L before getting two disconnected vertices. Ensuring that every pair of disconnected vertices has the same correlation is the far-enough property.

The first non-trivial pseudo-expectations arise from the Gaussian wave. To pick our Gaussians X_v, we will enforce that they satisfy the eigenvalue equation for the adjacency operator $A = A_{T_d}$.

$$\sum_{u \in N(v)} X_u = \lambda X_v$$

If we make this choice, we can easily compute the edge correlations

$$\lambda = \mathbb{E}[\lambda X_v^2] = \mathbb{E}\left[\sum_{u \sim v} X_u X_v\right] = d\,\mathbb{E}[X_u X_v]$$

Choosing the value $\lambda = -d\epsilon$, we get the desired edge correlations for our moment matrix. It turns out that we can pick such Gaussians for any λ as long as $|\lambda| \leq d$ [8]. This would give us an $\epsilon = 1$! For that, we would color vertices alternating colors on the whole tree. But how will we transfer that to the finite graph?

This is where we need to enforce the L-configuration-symmetry. As it turns out, for any λ with $|\lambda| \leq 2\sqrt{d-1}$, we can also construct a sequence of Gaussian processes converging to the Gaussian wave. Each process in the sequence is formed by taking a finite linear combination of i.i.d Gaussians at each vertex [8]. That limiting sequence is what Dembo et al. [10] use in their construction of the pseudo-expectation.

We can also make a simpler choice for our coefficients and arrive at an identical bound. Let us assign an independent standard Gaussian Y_v to each vertex $v \in T_d$. Our final Gaussians will be

$$X_v = \frac{1}{\zeta} \sum_{i=0}^{L} c^i \sum_{u : d(u,v) = i} Y_u$$

for some constants $c = -1/\sqrt{d-1}$ and ζ. The initial Gaussians Y_v are i.i.d standard Gaussians, so their covariance is $\mathbb{E}[Y_a Y_b] = 1(a = b)$. Therefore, we only get a contribution in $\mathbb{E}[X_a X_b]$ for each vertex c at distance at most L from both a and b.

We can figure out the normalization by finding the variance

$$\begin{aligned}
\mathbb{E}[X_v^2] &= \frac{1}{\zeta^2} \sum_{i=0}^{L} \sum_{u : d(u,v) = i} c^{2i}\,\mathbb{E}[Y_a Y_b] \\
&= \frac{1}{\zeta^2}\left(1 + \sum_{i=1}^{L} c^{2i} \cdot d(d-1)^{i-1}\right) \\
&= \left(1 + c^2 d(L-1)\right)/\zeta^2 \\
&= (c^2 dL - 1/(d-1))/\zeta^2
\end{aligned}$$

Because we want $\mathbb{E}[X_v^2] = 1$, this defines the value of ζ.

Now, we can compute the edge correlation $-\epsilon$ for an edge (u, v). For this, we need to count the number of nodes at distance a from u and distance b from v. Fortunately, every node is either closer to v and $d(v, x) = d(u, x) + 1$ or closer to u and $d(v, x) + 1 = d(u, x)$. Starting from vertex u, there are always $d - 1$ ways to step away from v and 1 way to step closer. Thus, the number of nodes at distance a from u and $a + 1$ from v is $(d-1)^a$.

$$\mathbb{E}[X_u X_v] = \frac{1}{\zeta^2} \cdot 2 \sum_{a=0}^{L-1} c^{2a+1}(d-1)^a = \frac{1}{\zeta^2} \cdot 2cL$$

into which we can plug our known value of ζ to find

$$\epsilon = -\mathbb{E}[X_u X_v] = -\frac{2cL}{c^2 dL - 1/(d-1)} = -\frac{2}{cd} + \frac{2}{cd(dL-1)} = \frac{2\sqrt{d-1}}{d} - O(1/L)$$

Note that for vertices at distance $> 2L$ there is 0 correlation, so we satisfy $(2L+1)$-configuration-symmetry.

In any case, we can easily transfer these moments down to the finite graph as long as the finite graph G has girth $g(G) > 2L+1$. To make the transfer, repeat the process of picking i.i.d Gaussians and taking their linear combinations. Because our calculations of ϵ only relied on there not being cycles within distance L of a pair of adjacent vertices, girth $g(G) > 2L+1$ is sufficient to guarantee the same calculations occur on the finite graph.

One interesting note is, though this geometric coefficient sequence achieves the same bound as the Gaussian wave approximations, it does not converge to the Gaussian wave. Let us take a new decay rate c with $|c| < 1/\sqrt{d-1}$ and take the limit process where the distance $L \to \infty$. We can re-compute what the normalization ζ is

$$\begin{aligned} 1 = \mathbb{E}[X_v^2] &= \frac{1}{\zeta^2}\left(1 + \sum_{i=1}^{\infty} c^{2i} d(d-1)^{i-1}\right) \\ &= \frac{1}{\zeta^2}\left(1 + c^2 d/(1 - c^2(d-1))\right) \\ &= \frac{1}{\zeta^2}\frac{1+c^2}{1-c^2(d-1)} \end{aligned}$$

and use that to find the nearby correlations (for uv and vw an edge).

$$\begin{aligned} \sigma_1' = \mathbb{E}[X_u X_v] &= \frac{1}{\zeta^2} \cdot 2 \cdot \left(\sum_{i=0}^{\infty} c^{2i+1}(d-1)^i\right) = \frac{2c}{1+c^2} \\ \sigma_2' = \mathbb{E}[X_u X_w] &= \frac{1}{\zeta^2}\left(2 \cdot \sum_{i=0}^{\infty} c^{2i+2}(d-1)^i + c^2 + \sum_{i=0}^{\infty} c^{2i+4}(d-2)(d-1)^i\right) \\ &= \frac{1}{\zeta^2} \cdot \frac{2c^2 + c^2(1-c^2(d-1)) + c^4(d-2)}{1-c^2(d-1)} = \frac{3c^2 - c^4}{1+c^2} \end{aligned}$$

We used similar counting techniques as before to compute these. We can compare these correlations to the ones for the Gaussian wave [8]:

$$\sigma_1 = \frac{\lambda}{d} \qquad\qquad \sigma_2 = \frac{\lambda^2 - d}{d(d-1)}$$

Clearly, for the distributions to match we need the first and second moments to match: $\sigma_1' = \sigma_1$ and $\sigma_2' = \sigma_2$. If we pick $\lambda = 2cd/(1+c^2)$ to get $\sigma_1' = \sigma_1$, then the second moments do not match. What this brief computation tells us is that the different optimal degree-2 pseudo-expectations are actually different and not only found via different methods.

4.2 Never Look Back

Rather than looking at the infinite tree T_d, Banks et al. [4] make use of non-backtracking walks to define the moment matrix. Like the adjacency matrix A, we can define the non-backtracking walk matrices $A^{(t)}$. The (i, j) entry $A^{(t)}_{i,j}$ is the number of non-backtracking walks of length t from vertex i to vertex j. A non-backtracking walk is any walk that does not return to the immediately-previous vertex: $i \to j \to i$. In the high-girth situation, the entry $A^{(t)}_{i,j}$ becomes the indicator that the distance $d(i, j) = t$.

Using these non-backtracking walks, we would like to define the moment matrix as

$$M = \sum_{t=0}^{L} c_t A^{(t)}$$

for some constant L and constants $c_0, \ldots, c_L$. Note that this definition exactly captures configuration-symmetry for the moment matrix M. Because M is indexed by singletons, the cells U that we consider are pairs of vertices. The configuration $C_L(i, j)$ is either the path between i and j or a pair of disconnected vertices. As above, the path configurations of length t are exactly captured by $A^{(t)}$ (in the high-girth situation). Thus, we assign the same value to each path configuration and, as M contains no other terms, we assign 0 to the disconnected configuration.

To show positivity, we will eventually construct M as an explicit matrix square. To do this, we first decompose the non-backtracking matrices $A^{(t)}$ with a simple recursion. A non-backtracking walk of length t is a non-backtracking walk of length $t - 1$ plus one more step that does not backtrack. To compute this, we can find the number of free last steps and subtract out the ones that backtrack. When backtracking one step we effectively have a non-backtracking walk of length $t - 2$, but we could have arrived in one of many different ways—one for each neighbor.

$$\begin{aligned}
A^{(0)} &= I \\
A^{(1)} &= A \\
A^{(2)} &= A \cdot A^{(1)} - dI = A^2 - dI \\
A^{(t)} &= A \cdot A^{(t-1)} - (d-1)A^{(t-2)} && t \geq 3
\end{aligned}$$

By solving the recursion, we can see that $A^{(t)}$ can be represented by a polynomial of degree exactly t in the adjacency matrix A. Thus, any polynomial of degree t in A can be represented by a linear combination of the non-backtracking powers $A^{(0)}, \ldots, A^{(t)}$.

To define the matrix M, we will make use of the roots of a specific polynomial $q_m(z)$ of degree $m = (L + 2)/2$ defined by Banks et al. [4]. Say the roots of $q_m(z)$ are $r_1 > \cdots > r_m$. Then, let our construction polynomial of degree L be

$$s(z) = \frac{1}{\zeta} \prod_{i=1}^{m-1} (z - r_i)^2$$

which is manifestly a square (and ζ is a normalization constant). We define the moment matrix $M = s(A) \succeq 0$. By using a quadrature rule, Banks et al. [4] are able to show that, when represented as explicitly configuration-symmetric, this matrix $M = \sum_{t=0}^{L} c_t A^{(t)}$ satisfies scaling ($c_0 = 1$) and has an edge coefficient

$$\epsilon = -c_1 = \frac{2(1 - o_L(1))\sqrt{d-1}}{d}$$

They also show that this is the optimal ϵ obtainable for degree-2 psuedo-expectations.

Using this recipe with $L = 2$, we obtain the matrix polynomial

$$M = \frac{1}{2}\left(I - \frac{1}{\sqrt{2d}} \cdot A\right) = \left(\frac{1}{2} + \frac{d}{2d}\right) \cdot I - \frac{1}{\sqrt{d}} \cdot A + \frac{1}{2d}(A^2 - dI) = I - \frac{1}{\sqrt{d}}A + \frac{1}{2d}A^{(2)}$$

which gives us an edge-coefficient of $\epsilon = 1/\sqrt{d}$. This edge-coefficient does not beat $2P_*/\sqrt{d}$, but the moment matrix is simple to understand.

5 Weight Watchers: Fix Your Girth

What we would like to do is be able to take pseudo-expectations on graphs with high girth and turn them into psuedo-expectations on graphs with low girth. The main reason why the assumption of constantly-high girth is fine is the number of short cycles is actually quite small. We quantify this key idea in the following lemma.

Lemma 5.1. *If γ, d are constant and the graph $G \sim \mathcal{G}_{n,d}^{reg}$, then with probability at least $1 - p(n) = 1 - p$ there exists a set B of size $|B| = O(\lg(1/p)$ such that the girth $g(G - B) \geq \gamma$.*

Proof. To form B, we will select one vertex from each cycle of length less than γ. This forces the girth $g(G - B) \geq \gamma$ because any smaller cycles have been disrupted.

Let the random variable X_k be the number of cycles of length k in G. Then, the random variables $X_3, \ldots, X_{\gamma-1}$ are asymptotically independent random variables distributed $X_k \sim \text{Poisson}((d-1)^2/2k)$ [7]. If we let the random variable $X = \sum_{k=3}^{\gamma-1} X_k$, we can upper-bound the size $|B| \leq X$. Because the X_k are independent and Poisson, $X \sim \text{Poisson}(\lambda)$ where

$$\lambda = \sum_{k=3}^{\gamma-1} \frac{(d-1)^2}{2k}$$

is a constant. Let $m = \max(2e\lambda, \lg(1/p)) = O(\lg(1/p))$. Using a Chernoff bound, we find that

$$\Pr[X \geq m] \leq e^{-\lambda} \cdot (e\lambda/m)^m \leq 2^{-\lg(1/p)} = p$$ ☐

In addition to the key property that the number of cycles is small, we will also use a gadget that makes the high-girth graph construction easier.

Lemma 5.2. *For every degree $d \geq 3$ and girth $\gamma \geq 3$, there exists a d-regular graph G with girth $g(G) \geq \gamma$.*

Proof. This result follows directly from the more general results of Dahan [9]. ☐

Using the gadget and the key idea, we will almost embed our random graphs G into larger graphs G' of high girth. Then, we can directly map down the pseudo-expectation with a small loss coming from the modifications in the almost embedding.

Lemma 5.3. *Suppose that, given any d-regular graph G' with girth $g(G') \geq kL$ (where k is a constant), there exists a degree-r pseudo-expectation $\widetilde{\mathbb{E}}_{G'}$ with a correlation of*

$$\widetilde{\mathbb{E}}_{G'}[x_i x_j] = -\epsilon$$

on each edge $ij \in E(G')$ and $\widetilde{\mathbb{E}}_{G'}[x_i x_j] = 0$ for $d(i,j) \geq L$. Then, given a d-regular random graph $G \sim \mathcal{G}_{n,d}^{reg}$, except with probability at most $p = p(n)$ there exists a degree-r pseudo-exepctation $\widetilde{\mathbb{E}}_G$ satisfying

$$\frac{2}{nd} \sum_{ij \in E(G)} \widetilde{\mathbb{E}}_G[x_i x_j] = -\epsilon(1 - O(\lg(1/p) \cdot 1/n))$$

Proof. Let the random graph $G \sim \mathcal{G}_{n,d}^{\text{reg}}$. We will construct a new graph G' with girth $g(G') \geq kL$ that shares most of its structure with G.

By Lemma 5.2, we know that there exists a d-regular graph H with girth $g(H) \geq \max(kL, L)$. Let B' be a minimum set of vertices such that the girth $g(G - B') \geq kL$. By Lemma 5.1, we know that $|B'| \leq O_n(\lg(1/p))$ with high probability. Let B be the set of edges outgoing from vertices in B'. Clearly, the girth $g(G - B) = g(G - B') \geq kL$ as well.

Initially, set the graph $G' = G$. Let $ab \in H$ be an arbitrary edge. For each edge $uv \in B$, add a new copy of H to G' and swap the edges uv and ab as in Figure 3. That is, remove edges uv, ab and add edges ua, vb. Clearly, this edge-swap process preserves the degree of each vertex; the graph G' at the end is d-regular.

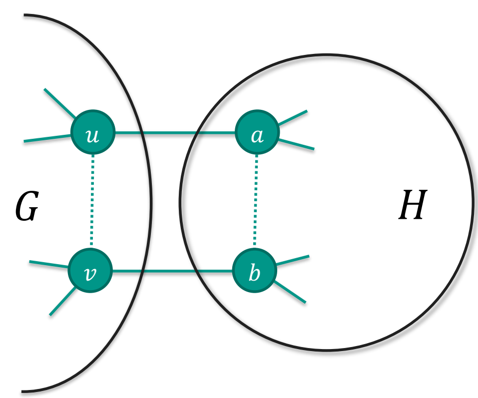

Figure 3: The edge-swap process

We would like to show that the girth $g(G') \geq kL$. Consider some cycle C in the graph G'. If C does not enter into any copies of H, it must be contained it $G - B$. By the girth property $g(G - B) \geq kL$, we know that $|C| \geq kL$. If C is entirely contained within some copy of H, we trivially know $|C| \geq kL$ by construction of H.

Otherwise, the cycle C intersects both G and some copies of H as in Figure 4. Consider one of those copies of H and look at the graph $C' = C \cap H$. Because the cycle C enters and

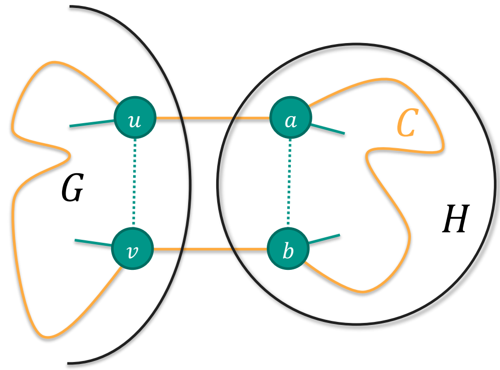

Figure 4: A cycle C in the graph G'

leaves a copy of H, the graph C' must be a path between a and b. Note that the graph C' cannot contain edge ab as the graph G' does not contain that edge. Thus, the cycle $C' + ab$ is a cycle in H and must satisfy $|C' + ab| \geq kL$; the original cycle satisfies $|C| \geq kL$. This same reasoning also concludes that any path between a vertex $u \in B'$ and a vertex $v \in G - \{u\}$ must go through a copy of the graph H and thus have length at least L.

By assumption, we have a pseudo-expectation $\widetilde{\mathbb{E}}_{G'}$ with average edge-correlation $-\epsilon$. Let M' be the moment matrix for $\widetilde{\mathbb{E}}_{G'}$. Restrict M' to those rows and columns that only contain vertices of G forming the matrix M. Because the matrix $M' \succeq 0$, the matrix $M \succeq 0$. Similarly, the poly-symmetry of M' implies poly-symmetry of M. Define $\widetilde{\mathbb{E}}_G$ by the moment matrix M. Note that this means $\widetilde{\mathbb{E}}_G[p(x)] = \widetilde{\mathbb{E}}_{G'}[p(x)]$ for any polynomial $p(x)$ only mentioning vertices in G.

To conclude the proof, we need to verify the average edge correlation. Because edges in B come from vertices in B', we know that any $ij \in B$ has $d_{G'}(i,j) \geq L$ and thus $\widetilde{\mathbb{E}}_G[x_i x_j] = 0$. Conversely, edges $ij \in G - B$ appear in G', so the coefficient $\widetilde{\mathbb{E}}_G[x_i x_j] = -\epsilon$.

$$
\begin{aligned}
\frac{2}{nd} \sum_{ij \in E(G)} \widetilde{\mathbb{E}}_G[x_i x_j] &= \frac{2}{nd} \left[\sum_{ij \in E(G)-B} \widetilde{\mathbb{E}}_G[x_i x_j] + \sum_{ij \in B} \widetilde{\mathbb{E}}_G[x_i x_j] \right] \\
&= \frac{2}{nd} \left[\sum_{ij \in E(G)-B} -\epsilon + \sum_{ij \in B} 0 \right] \\
&= -\epsilon \left(1 - \frac{2|B|}{nd} \right) = -\epsilon(1 - O(\lg(1/p) \cdot 1/n)) \qquad \square
\end{aligned}
$$

6 Extension Method

For making the pseudo-expectation, the typical strategy is to start with a candidate satisfying poly-symmetry for easy reasons and then work hard to show positivity. For the MaxCut degree-4 pseudo-expectation, we will be getting our candidate by extending the degree-2 pseudo-expectation.

To do this, we will look at the moment matrix indexed by ordered pairs of potentially duplicate vertices:

$$M = \widetilde{\mathbb{E}}[(x \otimes x) \cdot (x \otimes x)^{\intercal}]$$

We will naturally extend the notions of poly-symmetry and configuration-symmetry to this matrix M. For entry (a, b) of M with a and b being ordered pairs, the L-configuration is $C_L(a, b)$ and the polynomial (for poly-symmetry) is the multilinearization of $\prod_{i \in a} x_i \cdot \prod_{i \in b} x_i$. The plan will be to extend M into a full moment matrix by setting all of the odd-degree moments to 0. Note that M contains all of the even-degree moments.

Throughout this discussion, we will also think of the matrix M as describing weighted movement throughout the graph. Because M is indexed by ordered pairs, it will describe movement of pairs of vertices in the graph. For example, the matrix $A \otimes A$ describes moving each vertex independently along an edge as shown in Figure 5. We will think about the row index as the "start" or "left" pair of vertices and the column index as the "end" or "right" pair of vertices.

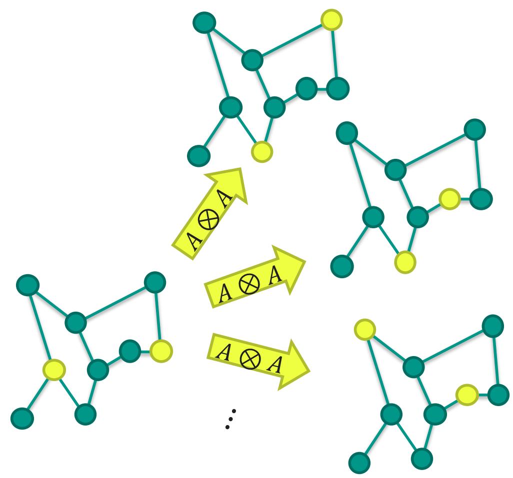

Figure 5: Movement according to the matrix $A \otimes A$

Let's define F to be our configuration-symmetric degree-2 solution: in the resulting pseudo-expectation, the matrix

$$F = \widetilde{\mathbb{E}}[x \cdot x^{\intercal}]$$

It would be beautiful if we could say $M = F \otimes F$, which means move each vertex in the pair independently according to F. Unfortunately, moving independently fails poly-symmetry. For example, we need

$$F_{a,c} \cdot F_{b,d} = M_{(a,b),(c,d)} = M_{(a,b),(d,c)} = F_{a,d} \cdot F_{b,c} \quad (1)$$
$$= M_{(a,c),(b,d)} = F_{a,b} \cdot F_{c,d} \quad (2)$$

as shown in Figure 6. However, we have no way of enforcing that our degree-2 solution F will satisfy these constraints. The real problem is that only using the tensor-product captures 8 of the 24 possible reorderings of 4 vertices.

To fix equality (1), we need to allow swapping the vertices before or after moving along $F \otimes F$. To do this, we will define a 0/1 swapping matrix $\Pi_{\neq}$ with the $(a, b), (c, d)$ entry

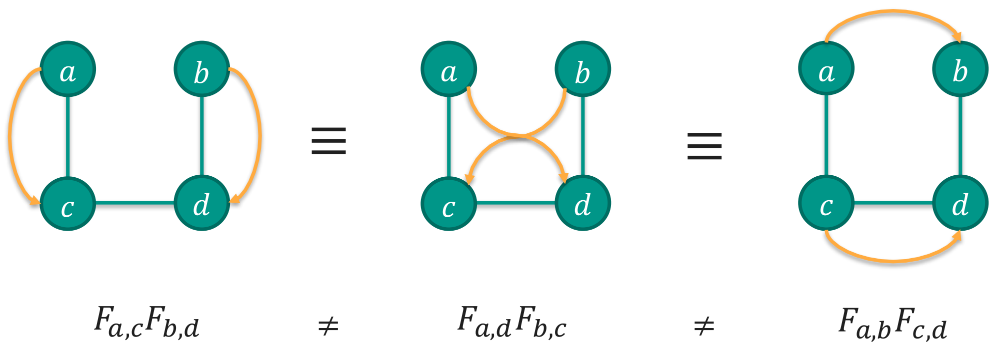

Figure 6: Requirements for poly-symmetry

non-zero if and only if $a \neq b$, $c \neq d$, and $\{a, b\} = \{c, d\}$. Then, we can use this matrix to enable swapping: $\Pi_{\neq} \cdot (F \otimes F) \cdot \Pi_{\neq}/2$. One can see how this improves the situation in Figure 7. Dividing by two cancels out the double-counting from the fact that swapping at the beginning and the end is the same as not swapping at all.

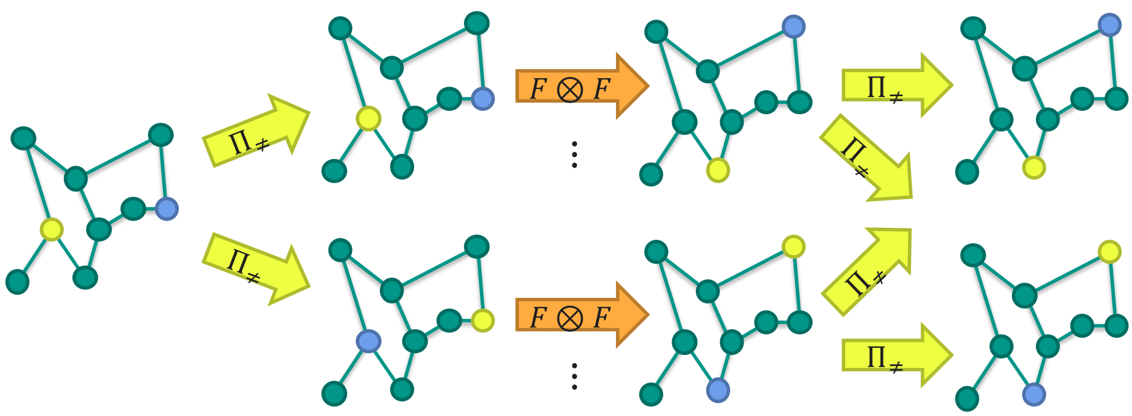

Figure 7: The action of $\Pi_{\neq}$

Next, we can fix equality (2) by allowing us to move from any F-related pair of vertices to any other F-related pair of vertices. Define $\text{vec}(F)_{(i,j)} = F_{i,j}$. In some sense, this vector serves as an indicator for vectors being related by F. Then, the outer product $\text{vec}(F) \cdot \text{vec}(F)^\intercal$ will allow moving from F-related vertices to F-related vertices, exactly how we want.

Putting these two fixes together, let's define our extension recipe as

$$\text{extend}(F) = \text{vec}(F) \cdot \text{vec}(F)^\intercal + \Pi_{\neq} \cdot (F \otimes F) \cdot \Pi_{\neq}/2$$

which almost works. Unfortunately, it has some glitches when the left and right vertex sets overlap (e.g. Figure 8). These glitches arise because the tensor product and outer product treat paths with overlapping vertices as disjoint paths, when, from the perspective of poly-symmetry, they should be treated as a combined path. Let us define the glitch matrix $G(F)$ as follows:

$$G(F)_{(a,b),(c,d)} = \begin{cases} 2F_{a,b}F_{c,d} & a \neq b, c \neq d, \{a, b\} \cap \{c, d\} \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

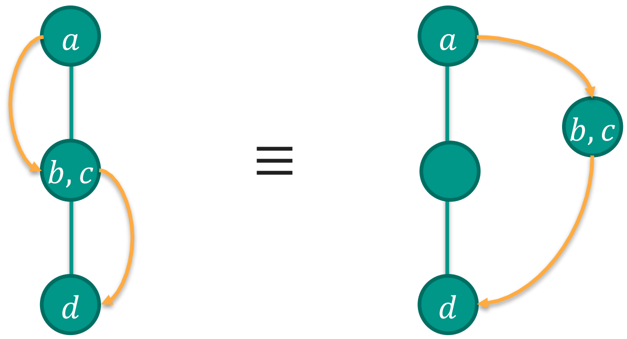

Figure 8: An example glitch. Arrows point from the LHS to the RHS

Using this glitch to fix the poly-symmetry issues, we get a valid matrix.

Lemma 6.1. *If F is poly-symmetric and L-configuration-symmetric then* $\text{extend}(F) - G(F)$ *is poly-symmetric and L-configuration-symmetric. Moreover, if an entry of* $\text{extend}(F) - G(F)$ *has polynomial $x_i x_j$, then that entry has value $F_{i,j}$.*

Proof. Label the parts of $M = \text{extend}(F) - G(F)$: $O = \text{vec}(F) \cdot \text{vec}(F)^\intercal$, $G = G(F)$, and $T = \Pi_{\neq} \cdot (F \otimes F) \cdot \Pi_{\neq}/2$.

Examine the entry $M_{(a,b),(c,d)} = e_{(a,b)}^\intercal M e_{(c,d)}$. The easiest term is $e_{(a,b)}^\intercal \cdot O \cdot e_{(c,d)} = F_{a,b} F_{c,d}$. For the rest, let's case on the polynomial $x_a x_b x_c x_d$.

- Case x_i^4. For poly-symmetry, we want the entry $M_{(i,i),(i,i)} = 1$. Because $a = b$, this term does not appear in G or T. Thus, $M_{(i,i),(i,i)} = O_{(i,i),(i,i)} = F_{i,i} F_{i,i} = 1$.

- Case $x_i^2 x_j^2$. For poly-symmetry, we'd also like to show this entry is 1. If $a = b$ then $c = d$ and the term only appears in O. Then, $M_{(i,i),(j,j)} = O_{(i,i),(j,j)} = F_{i,i} F_{j,j} = 1$.

  Otherwise, $a \neq b$ and $c \neq d$. The two sides share both vertices, so $G_{(a,b),(c,d)} = 2F_{a,b} F_{c,d}$. Because $\Pi_{\neq} \cdot e_{(c,d)} = e_{(c,d)} + e_{(d,c)}$ when $c \neq d$, we have

  $$e_{(a,b)}^\intercal \cdot (\Pi_{\neq} \cdot (F \otimes F) \cdot \Pi_{\neq}/2) \cdot e_{(c,d)} = (F_{a,c} F_{b,d} + F_{a,d} F_{b,c} + F_{b,c} F_{a,d} + F_{a,c} F_{b,d})/2 = F_{a,c} F_{b,d} + F_{a,d} F_{b,c}$$

  Let us suppose that $a = c$ and $b = d$; the other case is symmetric. Our total is

  $$\begin{aligned} M_{(a,b),(c,d)} &= F_{a,b} F_{c,d} + (F_{a,c} F_{b,d} + F_{a,d} F_{b,c}) - 2F_{a,b} F_{c,d} \\ &= F_{a,b} F_{a,b} + (F_{a,a} F_{b,b} + F_{a,b} F_{b,a}) - 2F_{a,b} F_{a,b} \\ &= F_{a,a} F_{b,b} = 1 \end{aligned}$$

- Case $x_i^3 x_j$. We need the entry to be $F_{i,j}$. Because F is configuration-symmetric, this type of entry will be as well (meaning it will match all other entries with the same configuration). By the pigeon-hole principle, either $a = b$ or $c = d$. This term can therefore only appear in O. By poly-symmetry of F, we have $F_{i,j} = F_{j,i}$. Thus, in every case the overall entry is $F_{i,i} F_{i,j} = F_{i,j}$.

- Case $x_k^2 x_i x_j$. We again need this entry to be $F_{i,j}$, which will be configuration-symmetric because F is. Suppose first that $a = b$ or $c = d$. Without loss of generality, we pick $a = b$. The only contribution is therefore O, so our value is $M_{(a,b),(c,d)} = O_{(a,b),(c,d)} = F_{a,b}F_{c,d} = F_{c,d} = F_{i,j}$.

  Otherwise, $a \neq b$ and $c \neq d$. Take the case when $a = c$; the other cases $a = d$, $b = c$, or $b = d$ are symmetric. Because the left and right sides share entries, G has a contribution. Like the $x_i^2 x_j^2$ case, our total will be

$$\begin{aligned} M_{(a,b),(c,d)} &= F_{a,b}F_{c,d} + (F_{a,c}F_{b,d} + F_{a,d}F_{b,c}) - 2F_{a,b}F_{c,d} \\ &= F_{k,b}F_{k,d} + (F_{k,k}F_{b,d} + F_{k,d}F_{b,k}) - 2F_{k,b}F_{k,d} \\ &= F_{k,k}F_{b,d} = F_{i,j} \end{aligned}$$

- Otherwise, all vertices are distinct. Because the two sides share no vertices, the glitch has no contribution. Our entry therefore will be

$$\begin{aligned} M_{(a,b),(c,d)} &= F_{a,b}F_{c,d} + (F_{a,c}F_{b,d} + F_{a,d}F_{b,c}) - 0 \\ &= F_{a,b}F_{c,d} + F_{a,c}F_{b,d} + F_{a,d}F_{c,b} \end{aligned}$$

  Notice by including all possible orderings (because F is poly-symmetric), this entry does not depend on the order of a, b, c, d. This entry is therefore poly-symmetric. Finally, because F is L-configuration-symmetric this term only depends on all pairs of distances between the vertices: their L-configuration. □

Next, let's transfer our matrix $M' = \text{extend}(F) - G(F)$ to the moment matrix M. We will assign

$$M = \begin{array}{r} {\scriptstyle S,T} \\ {\scriptstyle |S|\in\{0,2\}} \\ {\scriptstyle |S|=1} \end{array} \!\!\begin{array}{c} \begin{array}{cc} {\scriptstyle |T|\in\{0,2\}} & {\scriptstyle |T|=1} \end{array} \\ \begin{bmatrix} M^{(0,2)} & 0 \\ 0 & M^{(1)} \end{bmatrix} \end{array}$$

where $M^{(1)} = F$ and $M^{(0,2)}$ is a principal submatrix of M': choose the rows/columns $\emptyset \mapsto (0,0)$ and $\{x, y\} \mapsto (x, y)$ for $x < y$.

Lemma 6.2. *If F is poly- and configuration-symmetric, then the matrix M defined above is also poly- and configuration-symmetric.*

Proof. This result follows almost directly from Lemma 6.1. Let us consider a particular entry (S, T) of M. We handle the entry differently based on the size $|S \triangle T|$.

- Case $|S \triangle T|$ is odd. By construction, we have $M_{(S,T)} = 0$.

- Case $|S \triangle T| = 4$. The entry occurs entirely within $M^{(0,2)}$. By the symmetries of M', these entries all match.

- Case $|S \triangle T| = 0$. We know that $M_{(S,T)} = 1$ by symmetries of F and M'.

- Case $S \triangle T = \{i, j\}$. If $|S| \neq 1$ (and therefore $|T| \neq 1$), we have that $M^{(0,2)}_{S,T} = F_{i,j}$ as stated explicitly in Lemma 6.1.

  Otherwise, $M^{(1)}_{S,T} = F_{i,j}$ by poly-symmetry of F. Configuration-symmetry of F here again implies configuration-symmetry of M. □

6.1 Positivity

We believe that $M \succeq 0$ as defined but we have not yet been able to prove it. As such, we will instead show that $M\alpha + (1-\alpha)I \succeq 0$ for $\alpha \in [0,1]$ and a particular F, which will show that $M\alpha + (1-\alpha)I$ is valid.

Lemma 6.3. *Choosing $F = 1 + 1/\sqrt{d} \cdot A^{(1)} + 1/2d \cdot A^{(2)}$ and $\alpha = 1/6$, our moment matrix $M\alpha + (1-\alpha)I \succeq 0$.*

Proof. Recall that this F is the degree-2 polynomial solution from Banks et al. [4]. To show that $M\alpha + (1-\alpha) \succeq 0$, it then suffices to show that $M^{(0,2)}\alpha + (1-\alpha) \succeq 0$.

Clearly, $\text{vec}(F) \cdot \text{vec}(F)^\intercal \succeq 0$. Because $F \succeq 0$, we know that $F \otimes F \succeq 0$. This implies that $\Pi_{\neq} \cdot (F \otimes F) \cdot \Pi_{\neq}/2 \succeq 0$ because $\Pi_{\neq} = \Pi_{\neq}^\intercal$. If we let G be the submatrix of $G(F)$ in $M^{(0,2)}$, then we can write $M^{(0,2)} = P - G$ where $P \succeq 0$ (as the principal submatrix of $\text{extend}(F) \succeq 0$). With $\alpha = 1/6$, it will suffice to show that $\text{spr}(G) \leq 5$.

For this, we will use the trace method. Let us briefly consider what the entries of G are. For a non-zero entry (S,R) we know that $|S| = |R| = 2$ and that $S \cap R \neq \emptyset$. At that point, the value $G_{S,R} = 2F_S F_T$ (where we abuse notation with $F_{\{i,j\}} = F_{i,j}$). For a pair S to be related by F, the vertices must either be at distance 1 or at distance 2. To move from S to R, we pay the cost F_T and also the cost F_S.

Let $P = (P_0, P_1, \ldots, P_T)$ be a path where each entry $P_i \subseteq V$ is a set of size 0 or 2. The trace then is

$$\text{Tr}[G^T] = \sum_{P : P_0 = P_T} \prod_{i=1}^{T} G_{P_{i-1}, P_i}$$

Let's consider a particular path P. We can assume that none of the P_i have size zero because otherwise the product would be zero. Then, the total weight of that path would be

$$\prod_{i=1}^{T} G_{P_{i-1},P_i} = \prod_{i=1}^{T} 2F_{P_{i-1}} F_{P_i} = F_{P_0} \cdot \prod_{i=1}^{T} 2F_{P_i}^2 \cdot (1/F_{P_T}) = \prod_{i=1}^{T} 2F_{P_i}^2$$

where the last step follows if $F_{P_0} \neq 0$ because $P_0 = P_T$. Say that the path P is good if $P_0 = P_T$ and $F_{P_0} \neq 0$. Plugging back into the trace

$$\text{Tr}[G^T] = \sum_{P \text{ good}} \prod_{i=1}^{T} 2F_{P_i}^2 = \text{Tr}[H^T]$$

where we define

$$H_{S,T} = \begin{cases} 2F_T^2 & |S| = |T| = 2, S \cap T \neq 0, F_S \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

From any given pair of vertices S, there are at most $2d$ pairs T with $d(T) = 1$. Similarly, there are at most $2d(d-1) \leq 2d^2$ pairs T with $d(T) = 2$. Thus, the sum of magnitudes of any row of H is at most $2d \cdot 2 \cdot (1/\sqrt{d})^2 + 2d^2 \cdot 2 \cdot (1/2d)^2 \leq 5$. By diagonal dominance, we find

$$\text{Tr}[G^T]^{1/T} = \text{Tr}[H^T]^{1/T} \leq \left(\left(\binom{n}{2} + 1\right) \cdot 5^T\right)^{1/T} \leq 5 + \delta$$

for any $\delta > 0$. We conclude $\text{spr}(G) \leq 5$. □

We now have a valid degree-4 pseudo-expectation defined by the moment matrix $M/6 + 5/6$. What kind of ϵ does this give us? The coefficient on edges in the chosen F was $-1/\sqrt{d}$, but we scaled that down with $\alpha = 1/6$. Thus, our final edge coefficient is $\epsilon = 1/(6\sqrt{d})$. Unfortunately, this does not beat the target $2P_*/\sqrt{d}$, so our pseudo-expectation is not good enough to give a non-trivial SOS lower-bound.

7 Conclusion

The goal of this work was to find a non-trivial degree-4 SOS lower-bound for average-case MaxCut. We unfortunately did not achieve that goal because we were unable to completely analyze the spectrum of our constructed moment matrix. However, we identified a key property that we believe the pseudo-expectation should satisfy: configuration-symmetry. This configuration-symmetry arises naturally from our graph distribution, and drastically limits the search space for possible pseudo-expectations. We also showed how, instead of focusing on any random graphs, we can focus on graphs of high girth without much loss in the objective value. Using configuration-symmetry as our guide, we developed a recipe for extending degree-2 pseudo-expectations into degree-4 pseudo-expectations on graphs of high girth. We believe this recipe could produce a non-trivial degree-4 pseudo-expectation with more careful analysis.

References

[1] Martin Andersen, L. Vandenberghe, and J. Dahl. *CVXOPT: A Python package for convex optimization, version 1.2.2*, 2018. URL https://cvxopt.org/.

[2] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.*, 2019. URL http://docs.mosek.com/9.0/toolbox/index.html.

[3] Jess Banks and Luca Trevisan. Vector colorings of random, ramanujan, and large-girth irregular graphs. *CoRR*, abs/1907.02539, 2019. URL http://arxiv.org/abs/1907.02539.

[4] Jess Banks, Robert Kleinberg, and Cristopher Moore. The lovász theta function for random regular graphs and community detection in the hard regime. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2017, August 16-18, 2017, Berkeley, CA, USA*, pages 28:1–28:22, 2017. doi: 10.4230/LIPIcs.APPROX-RANDOM.2017.28. URL https://doi.org/10.4230/LIPIcs.APPROX-RANDOM.2017.28.

[5] Boaz Barak and David Steurer. Sum-of-squares proofs and the quest toward optimal algorithms. *Electronic Colloquium on Computational Complexity (ECCC)*, 21:59, 2014. URL http://eccc.hpi-web.de/report/2014/059.

[6] Boaz Barak, Samuel B. Hopkins, Jonathan A. Kelner, Pravesh Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique

problem. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 428–437, 2016. doi: 10.1109/FOCS.2016.53. URL https://doi.org/10.1109/FOCS.2016.53.

[7] Béla Bollobás. A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *Eur. J. Comb.*, 1(4):311–316, 1980. doi: 10.1016/S0195-6698(80)80030-8. URL https://doi.org/10.1016/S0195-6698(80)80030-8.

[8] Endre Csóka, Balázs Gerencsér, Viktor Harangi, and Bálint Virág. Invariant gaussian processes and independent sets on regular graphs of large girth. *Random Struct. Algorithms*, 47(2):284–303, 2015. doi: 10.1002/rsa.20547. URL https://doi.org/10.1002/rsa.20547.

[9] Xavier Dahan. Regular graphs of large girth and arbitrary degree. *Combinatorica*, 34 (4):407–426, 2014. doi: 10.1007/s00493-014-2897-6. URL https://doi.org/10.1007/s00493-014-2897-6.

[10] Amir Dembo, Andrea Montanari, and Subhabrata Sen. Extremal cuts of sparse random graphs. *CoRR*, abs/1503.03923, 2015. URL http://arxiv.org/abs/1503.03923.

[11] Yash Deshpande, Andrea Montanari, Ryan O'Donnell, Tselil Schramm, and Subhabrata Sen. The threshold for sdp-refutation of random regular NAE-3SAT. *CoRR*, abs/1804.05230, 2018. URL http://arxiv.org/abs/1804.05230.

[12] Joel Friedman, Jeff Kahn, and Endre Szemerédi. On the second eigenvalue in random regular graphs. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing, May 14-17, 1989, Seattle, Washigton, USA*, pages 587–598, 1989. doi: 10.1145/73007.73063. URL https://doi.org/10.1145/73007.73063.

[13] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42 (6):1115–1145, 1995. doi: 10.1145/227683.227684. URL https://doi.org/10.1145/227683.227684.

[14] Dmitriy Kunisky and Afonso S. Bandeira. A tight degree 4 sum-of-squares lower bound for the sherrington-kirkpatrick hamiltonian. *CoRR*, abs/1907.11686, 2019. URL http://arxiv.org/abs/1907.11686.

[15] J. Löfberg. Yalmip: A toolbox for modeling and optimization in matlab. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[16] Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, {II}. *Journal of Symbolic Computation*, 60(0):94 – 112, 2014. ISSN 0747-7171. doi: http://dx.doi.org/10.1016/j.jsc.2013.09.003. URL http://www.sciencedirect.com/science/article/pii/S0747717113001193.

[17] Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103, January 2017. ISSN 2376-5992. doi: 10.7717/peerj-cs.103. URL https://doi.org/10.7717/peerj-cs.103.

[18] Andrea Montanari. Optimization of the Sherrington-Kirkpatrick Hamiltonian. *arXiv e-prints*, art. arXiv:1812.10897, Dec 2018.

[19] Andrea Montanari and Subhabrata Sen. Semidefinite programs on sparse random graphs and their application to community detection. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 814–827, 2016. doi: 10.1145/2897518.2897548. URL http://doi.acm.org/10.1145/2897518.2897548.

[20] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014. ISBN 978-1-10-703832-5. URL http://www.cambridge.org/de/academic/subjects/computer-science/algorithmics-complexity-computer-algebra-and-computational-g/analysis-boolean-functions.

[21] Prasad Raghavendra. Optimal algorithms and inapproximability results for every csp? In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 245–254, 2008. doi: 10.1145/1374376.1374414. URL https://doi.org/10.1145/1374376.1374414.

[22] N. C. Wormald. *Models of Random Regular Graphs*, pages 239–298. London Mathematical Society Lecture Note Series. Cambridge University Press, 1999. doi: 10.1017/CBO9780511721335.010.