

## 8 Approximation Algorithms and Max-Cut

### 8.1 The Max-Cut problem

Unless the widely believed  $P \neq NP$  conjecture is false, there is no polynomial algorithm that can solve all instances of an NP-hard problem. Thus, when faced with an NP-hard problem (such as the Max-Cut problem discussed below) one has three options: to use an exponential type algorithm that solves exactly the problem in all instances, to design polynomial time algorithms that only work for some of the instances (hopefully the typical ones!), or to design polynomial algorithms that, in any instance, produce guaranteed approximate solutions. This section is about the third option. The second is discussed in later in the course, in the context of community detection.

The Max-Cut problem is the following: Given a graph  $G = (V, E)$  with non-negative weights  $w_{ij}$  on the edges, find a set  $S \subset V$  for which  $\text{cut}(S)$  is maximal. Goemans and Williamson [GW95] introduced an approximation algorithm that runs in polynomial time and has a randomized component to it, and is able to obtain a cut whose expected value is guaranteed to be no smaller than a particular constant  $\alpha_{GW}$  times the optimum cut. The constant  $\alpha_{GW}$  is referred to as the approximation ratio.

Let  $V = \{1, \dots, n\}$ . One can restate Max-Cut as

$$\begin{aligned} \max \quad & \frac{1}{2} \sum_{i < j} w_{ij} (1 - y_i y_j) \\ \text{s.t.} \quad & |y_i| = 1 \end{aligned} \tag{66}$$

The  $y_i$ 's are binary variables that indicate set membership, i.e.,  $y_i = 1$  if  $i \in S$  and  $y_i = -1$  otherwise.

Consider the following relaxation of this problem:

$$\begin{aligned} \max \quad & \frac{1}{2} \sum_{i < j} w_{ij} (1 - u_i^T u_j) \\ \text{s.t.} \quad & u_i \in \mathbb{R}^n, \|u_i\| = 1. \end{aligned} \tag{67}$$

This is in fact a relaxation because if we restrict  $u_i$  to be a multiple of  $e_1$ , the first element of the canonical basis, then (67) is equivalent to (66). For this to be a useful approach, the following two properties should hold:

- (a) Problem (67) needs to be easy to solve.
- (b) The solution of (67) needs to be, in some way, related to the solution of (66).

**Definition 8.1** *Given a graph  $G$ , we define  $\text{MaxCut}(G)$  as the optimal value of (66) and  $\mathcal{R}\text{MaxCut}(G)$  as the optimal value of (67).*

We start with property (a). Set  $X$  to be the Gram matrix of  $u_1, \dots, u_n$ , that is,  $X = U^T U$  where the  $i$ 'th column of  $U$  is  $u_i$ . We can rewrite the objective function of the relaxed problem as

$$\frac{1}{2} \sum_{i < j} w_{ij} (1 - X_{ij})$$

One can exploit the fact that  $X$  having a decomposition of the form  $X = Y^T Y$  is equivalent to being positive semidefinite, denoted  $X \succeq 0$ . The set of PSD matrices is a convex set. Also, the constraint

$\|u_i\| = 1$  can be expressed as  $X_{ii} = 1$ . This means that the relaxed problem is equivalent to the following semidefinite program (SDP):

$$\begin{aligned} \max \quad & \frac{1}{2} \sum_{i < j} w_{ij} (1 - X_{ij}) \\ \text{s.t.} \quad & X \succeq 0 \text{ and } X_{ii} = 1, \quad i = 1, \dots, n. \end{aligned} \tag{68}$$

This SDP can be solved (up to  $\epsilon$  accuracy) in time polynomial on the input size and  $\log(\epsilon^{-1})$  [VB96].

There is an alternative way of viewing (68) as a relaxation of (66). By taking  $X = yy^T$  one can formulate a problem equivalent to (66)

$$\begin{aligned} \max \quad & \frac{1}{2} \sum_{i < j} w_{ij} (1 - X_{ij}) \\ \text{s.t.} \quad & X \succeq 0, \quad X_{ii} = 1, \quad i = 1, \dots, n, \text{ and } \text{rank}(X) = 1. \end{aligned} \tag{69}$$

The SDP (68) can be regarded as a relaxation of (69) obtained by removing the non-convex rank constraint. In fact, this is how we will later formulate a similar relaxation for the minimum bisection problem.

We now turn to property (b), and consider the problem of forming a solution to (66) from a solution to (68). From the solution  $\{u_i\}_{i=1, \dots, n}$  of the relaxed problem (68), we produce a cut subset  $S'$  by randomly picking a vector  $r \in \mathbb{R}^n$  from the uniform distribution on the unit sphere and setting

$$S' = \{i \mid r^T u_i \geq 0\}$$

In other words, we separate the vectors  $u_1, \dots, u_n$  by a random hyperplane (perpendicular to  $r$ ). We will show that the cut given by the set  $S'$  is comparable to the optimal one.

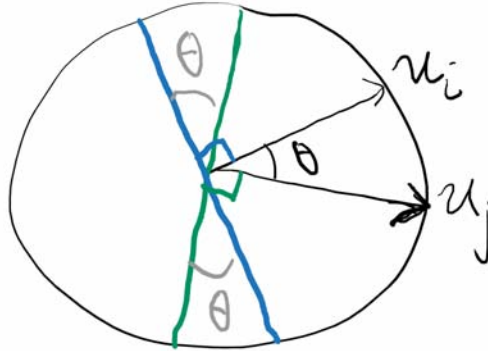


Figure 20:  $\theta = \arccos(u_i^T u_j)$

Let  $W$  be the value of the cut produced by the procedure described above. Note that  $W$  is a random variable, whose expectation is easily seen (see Figure 20) to be given by

$$\begin{aligned} \mathbb{E}[W] &= \sum_{i < j} w_{ij} \Pr \{ \text{sign}(r^T u_i) \neq \text{sign}(r^T u_j) \} \\ &= \sum_{i < j} w_{ij} \frac{1}{\pi} \arccos(u_i^T u_j). \end{aligned}$$

If we define  $\alpha_{GW}$  as

$$\alpha_{GW} = \min_{-1 \leq x \leq 1} \frac{\frac{1}{\pi} \arccos(x)}{\frac{1}{2}(1-x)},$$

it can be shown that  $\alpha_{GW} > 0.87$ .

It is then clear that

$$\mathbb{E}[W] = \sum_{i < j} w_{ij} \frac{1}{\pi} \arccos(u_i^T u_j) \geq \alpha_{GW} \frac{1}{2} \sum_{i < j} w_{ij} (1 - u_i^T u_j). \quad (70)$$

Let  $\text{MaxCut}(G)$  be the maximum cut of  $G$ , meaning the maximum of the original problem (66). Naturally, the optimal value of (77) is larger or equal than  $\text{MaxCut}(G)$ . Hence, an algorithm that solves (77) and uses the random rounding procedure described above produces a cut  $W$  satisfying

$$\text{MaxCut}(G) \geq \mathbb{E}[W] \geq \alpha_{GW} \frac{1}{2} \sum_{i < j} w_{ij} (1 - u_i^T u_j) \geq \alpha_{GW} \text{MaxCut}(G), \quad (71)$$

thus having an approximation ratio (in expectation) of  $\alpha_{GW}$ . Note that one can run the randomized rounding procedure several times and select the best cut.

Note that the above gives

$$\text{MaxCut}(G) \geq \mathbb{E}[W] \geq \alpha_{GW} \mathcal{R} \text{MaxCut}(G) \geq \alpha_{GW} \text{MaxCut}(G)$$

## 8.2 Can $\alpha_{GW}$ be improved?

A natural question is to ask whether there exists a polynomial time algorithm that has an approximation ratio better than  $\alpha_{GW}$ .

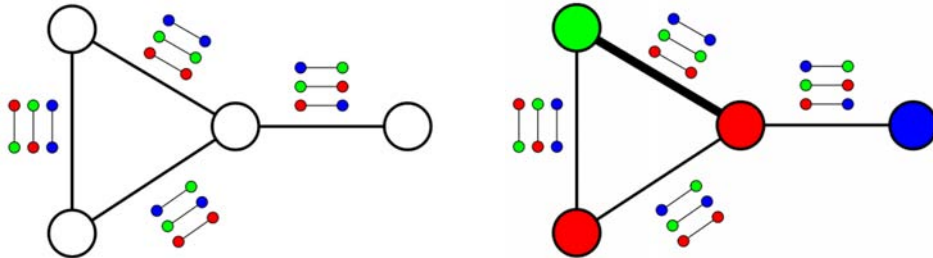


Figure 21: The Unique Games Problem

© Thore Husfeldt. All rights reserved. This content is excluded from our Creative Commons license. For more information, see <http://ocw.mit.edu/help/faq-fair-use/>.

The unique games problem (as depicted in Figure 21) is the following: Given a graph and a set of  $k$  colors, and, for each edge, a matching between the colors, the goal in the unique games problem is to color the vertices as to agree with as high of a fraction of the edge matchings as possible. For example, in Figure 21 the coloring agrees with  $\frac{3}{4}$  of the edge constraints, and it is easy to see that one cannot do better.

The Unique Games Conjecture of Khot [Kho02], has played a major role in hardness of approximation results.

**Conjecture 8.2** *For any  $\epsilon > 0$ , the problem of distinguishing whether an instance of the Unique Games Problem is such that it is possible to agree with a  $\geq 1 - \epsilon$  fraction of the constraints or it is not possible to even agree with a  $\epsilon$  fraction of them, is NP-hard.*

There is a sub-exponential time algorithm capable of distinguishing such instances of the unique games problem [ABS10], however no polynomial time algorithm has been found so far. At the moment one of the strongest candidates to break the Unique Games Conjecture is a relaxation based on the Sum-of-squares hierarchy that we will discuss below.

**Open Problem 8.1** *Is the Unique Games conjecture true? In particular, can it be refuted by a constant degree Sum-of-squares relaxation?*

Remarkably, approximating **Max-Cut** with an approximation ratio better than  $\alpha_{GW}$  is as hard as refuting the Unique Games Conjecture (UG-hard) [KKMO05]. More generally, if the Unique Games Conjecture is true, the semidefinite programming approach described above produces optimal approximation ratios for a large class of problems [Rag08].

Not depending on the Unique Games Conjecture, there is a NP-hardness of approximation of  $\frac{16}{17}$  for **Max-Cut** [Has02].

**Remark 8.3** *Note that a simple greedy method that assigns membership to each vertex as to maximize the number of edges cut involving vertices already assigned achieves an approximation ratio of  $\frac{1}{2}$  (even of  $\frac{1}{2}$  of the total number of edges, not just of the optimal cut).*

### 8.3 A Sums-of-Squares interpretation

We now give a different interpretation to the approximation ratio obtained above. Let us first slightly reformulate the problem (recall that  $w_{ii} = 0$ ).

$$\begin{aligned}
\max_{y_i = \pm 1} \frac{1}{2} \sum_{i < j} w_{ij} (1 - y_i y_j) &= \max_{y_i = \pm 1} \frac{1}{4} \sum_{i, j} w_{ij} (1 - y_i y_j) \\
&= \max_{y_i = \pm 1} \frac{1}{4} \sum_{i, j} w_{ij} \left( \frac{y_i^2 + y_j^2}{2} - y_i y_j \right) \\
&= \max_{y_i = \pm 1} \frac{1}{4} \left( - \sum_{i, j} w_{ij} y_i y_j + \frac{1}{2} \sum_i \left[ \sum_j w_{ij} \right] y_i^2 + \frac{1}{2} \sum_j \left[ \sum_i w_{ij} \right] y_j^2 \right) \\
&= \max_{y_i = \pm 1} \frac{1}{4} \left( - \sum_{i, j} w_{ij} y_i y_j + \frac{1}{2} \sum_i \deg(i) y_i^2 + \frac{1}{2} \sum_j \deg(j) y_j^2 \right) \\
&= \max_{y_i = \pm 1} \frac{1}{4} \left( - \sum_{i, j} w_{ij} y_i y_j + \sum_i \deg(i) y_i^2 \right) \\
&= \max_{y_i = \pm 1} \frac{1}{4} y^T L_G y,
\end{aligned}$$

where  $L_G = D_G - W$  is the Laplacian matrix,  $D_G$  is a diagonal matrix with  $(D_G)_{ii} = \deg(i) = \sum_j w_{ij}$  and  $W_{ij} = w_{ij}$ .

This means that we rewrite (66) as

$$\begin{aligned} \max \quad & \frac{1}{4} y^T L_G y \\ & y_i = \pm 1, \quad i = 1, \dots, n. \end{aligned} \tag{72}$$

Similarly, (68) can be written (by taking  $X = yy^T$ ) as

$$\begin{aligned} \max \quad & \frac{1}{4} \text{Tr}(L_G X) \\ \text{s.t.} \quad & X \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \dots, n. \end{aligned} \tag{73}$$

Indeed, given

Next lecture we derive the formulation of the dual program to (73) in the context of recovery in the Stochastic Block Model. Here we will simply show weak duality. The dual is given by

$$\begin{aligned} \min \quad & \text{Tr}(D) \\ \text{s.t.} \quad & D \text{ is a diagonal matrix} \\ & D - \frac{1}{4} L_G \succeq 0. \end{aligned} \tag{74}$$

Indeed, if  $X$  is a feasible solution to (73) and  $D$  a feasible solution to (74) then, since  $X$  and  $D - \frac{1}{4} L_G$  are both positive semidefinite  $\text{Tr} \left[ X \left( D - \frac{1}{4} L_G \right) \right] \geq 0$  which gives

$$0 \leq \text{Tr} \left[ X \left( D - \frac{1}{4} L_G \right) \right] = \text{Tr}(XD) - \frac{1}{4} \text{Tr}(L_G X) = \text{Tr}(D) - \frac{1}{4} \text{Tr}(L_G X),$$

since  $D$  is diagonal and  $X_{ii} = 1$ . This shows weak duality, the fact that the value of (74) is larger than the one of (73).

If certain conditions, the so called Slater conditions [VB04, VB96], are satisfied then the optimal values of both programs are known to coincide, this is known as strong duality. In this case, the Slater conditions ask whether there is a matrix strictly positive definite that is feasible for (73) and the identity is such a matrix. This means that there exists  $D^\natural$  feasible for (74) such that

$$\text{Tr}(D^\natural) = \mathcal{R}\text{MaxCut}.$$

Hence, for any  $y \in \mathbb{R}^n$  we have

$$\frac{1}{4} y^T L_G y = \mathcal{R}\text{MaxCut} - y^T \left( D^\natural - \frac{1}{4} L_G \right)^T + \sum_{i=1}^n D_{ii} (y_i^2 - 1). \tag{75}$$

Note that (75) certifies that no cut of  $G$  is larger than  $\mathcal{R}\text{MaxCut}$ . Indeed, if  $y \in \{\pm 1\}^2$  then  $y_i^2 = 1$  and so

$$\mathcal{R}\text{MaxCut} - \frac{1}{4} y^T L_G y = y^T \left( D^\natural - \frac{1}{4} L_G \right)^T.$$

Since  $D^\natural - \frac{1}{4}L_G \succeq 0$ , there exists  $V$  such that  $D^\natural - \frac{1}{4}L_G = VV^T$  with the columns of  $V$  denoted by  $v_1, \dots, v_n$ . This means that meaning that  $y^T (D^\natural - \frac{1}{4}L_G)^T = \|V^T y\|^2 = \sum_{k=1}^n (v_k^T y)^2$ . This means that, for  $y \in \{\pm 1\}^2$ ,

$$\mathcal{R}\text{MaxCut} - \frac{1}{4}y^T L_G y = \sum_{k=1}^n (v_k^T y)^2.$$

In other words,  $\mathcal{R}\text{MaxCut} - \frac{1}{4}y^T L_G y$  is, in the hypercube ( $y \in \{\pm 1\}^2$ ) a sum-of-squares of degree 2. This is known as a sum-of-squares certificate [BS14, Bar14, Par00, Las01, Sho87, Nes00]; indeed, if a polynomial is a sum-of-squares naturally it is non-negative.

Note that, by definition,  $\text{MaxCut} - \frac{1}{4}y^T L_G y$  is always non-negative on the hypercube. This does not mean, however, that it needs to be a sum-of-squares<sup>33</sup> of degree 2.

(A Disclaimer: the next couple of paragraphs are a bit hand-wavy, they contain some of intuition for the Sum-of-squares hierarchy but for details and actual formulations, please see the references.)

The remarkable fact is that, if one bounds the degree of the sum-of-squares certificate, it can be found using Semidefinite programming [Par00, Las01]. In fact, SDPs (74) and (74) are finding the smallest real number  $\Lambda$  such that  $\Lambda - \frac{1}{4}y^T L_G y$  is a sum-of-squares of degree 2 over the hypercube, the dual SDP is finding a certificate as in (75) and the primal is constraining the moments of degree 2 of  $y$  of the form  $X_{ij} = y_i y_j$  (see [Bar14] for some nice lecture notes on Sum-of-Squares, see also Remark 8.4). This raises a natural question of whether, by allowing a sum-of-squares certificate of degree 4 (which corresponds to another, larger, SDP that involves all monomials of degree  $\leq 4$  [Bar14]) one can improve the approximation of  $\alpha_{GW}$  to Max-Cut. Remarkably this is open.

**Open Problem 8.2** 1. *What is the approximation ratio achieved by (or the integrality gap of) the Sum-of-squares degree 4 relaxation of the Max-Cut problem?*

2. *The relaxation described above (of degree 2) (74) is also known to produce a cut of  $1 - \mathcal{O}(\sqrt{\epsilon})$  when a cut of  $1 - \epsilon$  exists. Can the degree 4 relaxation improve over this?*

3. *What about other (constant) degree relaxations?*

**Remark 8.4 (triangular inequalities and Sum of squares level 4)** *A (simpler) natural question is whether the relaxation of degree 4 is actually strictly tighter than the one of degree 2 for Max-Cut (in the sense of forcing extra constraints). What follows is an interesting set of inequalities that degree 4 enforces and that degree 2 doesn't, known as triangular inequalities.*

Since  $y_i \in \{\pm 1\}$  we naturally have that, for all  $i, j, k$

$$y_i y_j + y_j y_k + y_k y_i \geq -1,$$

this would mean that, for  $X_{ij} = y_i y_j$  we would have,

$$X_{ij} + X_{jk} + X_{ik} \geq -1,$$

however it is not difficult to see that the SDP (73) of degree 2 is only able to constraint

$$X_{ij} + X_{jk} + X_{ik} \geq -\frac{3}{2},$$

---

<sup>33</sup>This is related with Hilbert's 17th problem [Sch12] and Stengle's Positivstellensatz [Ste74]

which is considerably weaker. There are a few different ways of thinking about this, one is that the three vector  $u_i, u_j, u_k$  in the relaxation may be at an angle of 120 degrees with each other. Another way of thinking about this is that the inequality  $y_i y_j + y_j y_k + y_k y_i \geq -\frac{3}{2}$  can be proven using sum-of-squares proof with degree 2:

$$(y_i + y_j + y_k)^2 \geq 0 \quad \Rightarrow \quad y_i y_j + y_j y_k + y_k y_i \geq -\frac{3}{2}$$

However, the stronger constraint cannot.

On the other hand, if degree 4 monomials are involved, (let's say  $X_S = \prod_{s \in S} y_s$ , note that  $X_\emptyset = 1$  and  $X_{ij} X_{ik} = X_{jk}$ ) then the constraint

$$\begin{bmatrix} X_\emptyset \\ X_{ij} \\ X_{jk} \\ X_{ki} \end{bmatrix} \begin{bmatrix} X_\emptyset \\ X_{ij} \\ X_{jk} \\ X_{ki} \end{bmatrix}^T = \begin{bmatrix} 1 & X_{ij} & X_{jk} & X_{ki} \\ X_{ij} & 1 & X_{ik} & X_{jk} \\ X_{jk} & X_{ik} & 1 & X_{ij} \\ X_{ki} & X_{jk} & X_{ij} & 1 \end{bmatrix} \succeq 0$$

implies  $X_{ij} + X_{jk} + X_{ik} \geq -1$  just by taking

$$\mathbf{1}^T \begin{bmatrix} 1 & X_{ij} & X_{jk} & X_{ki} \\ X_{ij} & 1 & X_{ik} & X_{jk} \\ X_{jk} & X_{ik} & 1 & X_{ij} \\ X_{ki} & X_{jk} & X_{ij} & 1 \end{bmatrix} \mathbf{1} \geq 0.$$

Also, note that the inequality  $y_i y_j + y_j y_k + y_k y_i \geq -1$  can indeed be proven using sum-of-squares proof with degree 4 (recall that  $y_i^2 = 1$ ):

$$(1 + y_i y_j + y_j y_k + y_k y_i)^2 \geq 0 \quad \Rightarrow \quad y_i y_j + y_j y_k + y_k y_i \geq -1.$$

Interestingly, it is known [KV13] that these extra inequalities alone will not increase the approximation power (in the worst case) of (68).

## 8.4 The Grothendieck Constant

There is a somewhat similar remarkable problem, known as the Grothendieck problem [AN04, AMMN05]. Given a matrix  $A \in \mathbb{R}^{n \times m}$  the goal is to maximize

$$\begin{aligned} \max \quad & x^T A y \\ \text{s.t.} \quad & x_i = \pm 1, \forall_i \\ & y_j = \pm 1, \forall_j \end{aligned} \tag{76}$$

Note that this is similar to problem (66). In fact, if  $A \succeq 0$  it is not difficult to see that the optimal solution of (76) satisfies  $y = x$  and so if  $A = L_G$ , since  $L_G \succeq 0$ , (76) reduces to (66). In fact, when  $A \succeq 0$  this problem is known as the little Grothendieck problem [AN04, CW04, BKS13a].

Even when  $A$  is not positive semidefinite, one can take  $z^T = [x^T \ y^T]$  and the objective can be written as

$$z^T \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix} z.$$

Similarly to the approximation ratio in Max-Cut, the Grothendieck constant [Pis11]  $K_G$  is the maximum ratio (over all matrices  $A$ ) between the SDP relaxation

$$\begin{aligned} \max \quad & \sum_{ij} A_{ij} u_i^T v_j \\ \text{s.t.} \quad & u_i \in \mathbb{R}^{n+m}, \|u_i\| = 1, \\ & v_j \in \mathbb{R}^{n+m}, \|v_j\| = 1 \end{aligned} \tag{77}$$

and 76, and its exact value is still unknown, the best known bounds are available here [] and are  $1.676 < K_G < \frac{\pi}{2 \log(1+\sqrt{2})}$ . See also page 21 here [F+14]. There is also a complex valued analogue [Haa87].

**Open Problem 8.3** *What is the real Grothendieck constant  $K_G$ ?*

## 8.5 The Paley Graph

Let  $p$  be a prime such that  $p \equiv 1 \pmod{4}$ . The Paley graph of order  $p$  is a graph on  $p$  nodes (each node associated with an element of  $\mathbb{Z}_p$ ) where  $(i, j)$  is an edge if  $i - j$  is a quadratic residue modulo  $p$ . In other words,  $(i, j)$  is an edge if there exists  $a$  such that  $a^2 \equiv i - j \pmod{p}$ . Let  $\omega(p)$  denote the clique number of the Paley graph of order  $p$ , meaning the size of its largest clique. It is conjectured that  $\omega(p) \lesssim \text{poly}(\log p)$  but the best known bound is  $\omega(p) \leq \sqrt{p}$  (which can be easily obtained). The only improvement to date is that, infinitely often,  $\omega(p) \leq \sqrt{p} - 1$ , see [BRM13].

The theta function of a graph is a Semidefinite programming based relaxation of the independence number [Lov79] (which is the clique number of the complement graph). As such, it provides an upper bound on the clique number. In fact, this upper bound for Paley graph matches  $\omega(p) \leq \sqrt{p}$ .

Similarly to the situation above, one can define a degree 4 sum-of-squares analogue to  $\theta(G)$  that, in principle, has the potential to giving better upper bounds. Indeed, numerical experiments in [GLV07] seem to suggest that this approach has the potential to improve on the upper bound  $\omega(p) \leq \sqrt{p}$ .

**Open Problem 8.4** *What are the asymptotics of the Paley Graph clique number  $\omega(p)$ ? Can the SOS degree 4 analogue of the theta number help upper bound it?* <sup>34</sup>

Interestingly, a polynomial improvement on Open Problem 6.4. is known to imply an improvement on this problem [BMM14].

## 8.6 An interesting conjecture regarding cuts and bisections

Given  $d$  and  $n$  let  $G^{reg}(n, d)$  be a random  $d$ -regular graph on  $n$  nodes, drawn from the uniform distribution on all such graphs. An interesting question is to understand the typical value of the Max-Cut such a graph. The next open problem is going to involve a similar quantity, the Maximum Bisection. Let  $n$  be even, the Maximum Bisection of a graph  $G$  on  $n$  nodes is

$$\text{MaxBis}(G) = \max_{S: |S| = \frac{n}{2}} \text{cut}(S),$$

and the related Minimum Bisection (which will play an important role in next lectures), is given by

$$\text{MinBis}(G) = \min_{S: |S| = \frac{n}{2}} \text{cut}(S),$$

---

<sup>34</sup>The author thanks Dustin G. Mixon for suggesting this problem.



A typical bisection will cut half the edges, meaning  $\frac{d}{4}n$ . It is not surprising that, for large  $n$ ,  $\text{MaxBis}(G)$  and  $\text{MinBis}(G)$  will both fluctuate around this value, the amazing conjecture [ZB09] is that their fluctuations are the same.

**Conjecture 8.5** ([ZB09]) *Let  $G \sim G^{\text{reg}}(n, d)$ , then for all  $d$ , as  $n$  grows*

$$\frac{1}{n} (\text{MaxBis}(G) + \text{MinBis}(G)) = \frac{d}{2} + o(1),$$

where  $o(1)$  is a term that goes to zero with  $n$ .

**Open Problem 8.5** *Prove or disprove Conjecture 8.5.*

Recently, it was shown that the conjecture holds up to  $o(\sqrt{d})$  terms [DMS15]. We also point the reader to this paper [Lyo14], that contains bounds that are meaningful already for  $d = 3$ .

MIT OpenCourseWare  
<http://ocw.mit.edu>

18.S096 Topics in Mathematics of Data Science  
Fall 2015

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.