Mildly Exponential Time Approximation Algorithms for Vertex Cover, Balanced Separator and Uniform Sparsest Cut

Pasin Manurangsi
University of California, Berkeley, USA
pasin@berkeley.edu

Luca Trevisan
University of California, Berkeley, USA
luca@berkeley.edu

Abstract

In this work, we study the trade-off between the running time of approximation algorithms and their approximation guarantees. By leveraging a structure of the "hard" instances of the Arora-Rao-Vazirani lemma [6, 42], we show that the Sum-of-Squares hierarchy can be adapted to provide "fast", but still exponential time, approximation algorithms for several problems in the regime where they are believed to be NP-hard. Specifically, our framework yields the following algorithms; here n denote the number of vertices of the graph and r can be any positive real number greater than 1 (possibly depending on n).

- A $\left(2 - \frac{1}{O(r)}\right)$-approximation algorithm for Vertex Cover that runs in $\exp\left(\frac{n}{2^{r^2}}\right) n^{O(1)}$ time.
- An $O(r)$-approximation algorithms for Uniform Sparsest Cut and Balanced Separator that runs in $\exp\left(\frac{n}{2^{r^2}}\right) n^{O(1)}$ time.

Our algorithm for Vertex Cover improves upon Bansal et al.'s algorithm [9] which achieves $\left(2 - \frac{1}{O(r)}\right)$-approximation in time $\exp\left(\frac{n}{r^r}\right) n^{O(1)}$. For Uniform Sparsest Cut and Balanced Separator, our algorithms improve upon $O(r)$-approximation $\exp\left(\frac{n}{2^r}\right) n^{O(1)}$-time algorithms that follow from a work of Charikar et al. [17].

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis

Keywords and phrases Approximation algorithms, Exponential-time algorithms, Vertex Cover, Sparsest Cut, Balanced Separator

Digital Object Identifier 10.4230/LIPIcs.APPROX-RANDOM.2018.20

Related Version A full version of the paper is available at `https://arxiv.org/abs/1807.09898`.

Funding This material is based upon work supported by the National Science under Grants No. CCF 1655215 and CCF 1815434.

1 Introduction

Approximation algorithms and fast (sub)exponential time exact algorithms are among the two most popular approaches employed to tackle NP-hard problems. While both have had their fair share of successes, they seem to hit roadblocks for a number of reasons; the PCP theorem [7, 5] and the theory of hardness of approximation developed from it have established, for many optimization problems, that trivial algorithms are the best one could hope for (in polynomial time). On the other hand, the Exponential Time Hypothesis (ETH) [30, 31] and the fine-grained reductions surrounding it have demonstrated that "brute force" algorithms are, or at least close to, the fastest possible for numerous natural problems.

These barriers have led to studies in the cross-fertilization between the two fields, in which one attempts to apply both techniques simultaneously to overcome known lower bounds. Generally speaking, these works study the trade-offs between the running time of the algorithms and the approximation ratio. In other words, a typical question arising here is: what is the best running time for an algorithm with a given approximation ratio τ?

Optimization problems often admit natural "limited brute force" approximation algorithms that use brute force to find the optimal solution restricted to a subset of variables and then extend this to a whole solution. Similar to the study of fast exact algorithms for which a general motivating question is whether one can gain a noticeable speedup over "brute force", the analogous question when dealing with approximation algorithms is whether one can do significantly better than these limited brute force algorithms.

For example, let us consider the E3SAT problem, which is to determine whether a given 3CNF formula is satisfiable. The brute force (exact) algorithm runs in $2^{O(n)}$ time, while ETH asserts that it requires $2^{\Omega(n)}$ time to solve the problem. The optimization version of E3SAT is the Max E3SAT problem, where the goal is to find an assignment that satisfies as many clauses as possible. On the purely approximation front, a trivial algorithm that assigns every variable uniformly independently at random gives 7/8-approximation for Max E3SAT, while Hastad's seminal work [29] established NP-hardness for obtaining $(7/8+\varepsilon)$-approximation for any constant $\varepsilon > 0$. The "limited brute force" algorithm for Max E3SAT chooses a subset of $O(\epsilon n)$ variables, enumerates all possible assignments to those variables and picks values of the remaining variables randomly; this achieves $(7/8+\varepsilon)$-approximation in time $2^{O(\epsilon n)}$. Interestingly, it is known that running time of $2^{\Omega(\text{poly}(\varepsilon)n)}$ is necessary to gain a $(7/8+\varepsilon)$-approximation if one uses Sum-of-Squares relaxations [26, 53, 40], which gives some evidence that the running time of "limited brute force" $(7/8+\varepsilon)$ approximation algorithms for Max E3SAT are close to best possible.

In contrast to Max E3SAT, one can do much better than "limited brute force" for Unique Games. Specifically, Arora et al. [3] show that one can satisfy an ϵ fraction of clauses in a $(1-\epsilon)$-satisfiable instance of Unique Games in time $2^{n/exp(1/\epsilon)}$, a significant improvement over the trivial $2^{O(\varepsilon n)}$ time "limited brute force" algorithm. This algorithm was later improved by the celebrated algorithm of Arora, Barak and Steurer [2] that runs in time $2^{n^{poly(\epsilon)}}$.

A number of approximation problems, such as $(2-\varepsilon)$-approximation of Vertex Cover [38, 10], $(0.878\cdots+\varepsilon)$ approximation of Max Cut [35], and constant approximation of Non-uniform Sparsest Cut [18, 39] are known to be at least as hard as Unique Games, but are not known to be equivalent to Unique Games. If they were equivalent, the subexponential algorithm of [2] would also extend to these other problems. It is then natural to ask whether these problems admit subexponential time algorithms, or at least "better than brute force" algorithms. Indeed, attempts have been made to design such algorithms [2, 27], although these algorithms only achieve significant speed-up for specific classes of instances, not all worst case instances.

Recently, Bansal et al. [9] presented a "better than brute force" algorithm for Vertex Cover, which achieve a $(2-1/O(r))$-approximation in time $2^{O(n/r^r)}$. Note that the trade-off between approximation and running time is more analogous to the [3] algorithm for Unique Games than with the "limited brute force" algorithm for Max 3ESAT discussed above.

The algorithm of Bansal et al. is partially combinatorial and is based on a reduction to the Vertex Cover problem in bounded-degree graphs, for which better approximation algorithms are known compared to general graphs. Curiously, the work of Bansal et al. does not subsume the best known polynomial time algorithm for vertex cover: Karakostas [32] shows that there is a polynomial time algorithm for vertex cover that achieves a $\left(2 - \frac{\Omega(1)}{\sqrt{\log n}}\right)$ approximation ratio, but if one set $r := \sqrt{\log n}$ in the algorithm of Bansal et al. one does not get a polynomial running time.

This overview raises a number of interesting questions: is it possible to replicate, or improve, the vertex cover approximation of Bansal et al. [9] using Sum-of-Square relaxations? A positive result would show that, in a precise sense, $(7/8 + \varepsilon)$ approximation of Max 3SAT is "harder" than $(2 - \varepsilon)$ approximation for Vertex Cover (since the former requires $\mathrm{poly}(\varepsilon) \cdot n$ rounds while the latter would be achievable with $n/\exp(1/\varepsilon)$ rounds). Is it possible to have a "better than brute force" approximation algorithm for vertex cover that recovers Karakostas's algorithm as a special case? Is it possible to do the same for other problems that are known to be Unique-Games-hard but not NP-hard, such as constant-factor approximation of balanced separator?

1.1 Our Results

In this work, we answer the above questions affirmatively by designing "fast" exponential time approximation algorithms for Vertex Cover, Uniform Sparsest Cut and (Uniform) Balanced Separator. For Vertex Cover, our algorithm gives $(2 - 1/O(r))$-approximation in time $\exp(n/2^{r^2})n^{O(1)}$ where n is the number of vertices in the input graph and r is a parameter that can be any real number at least one (and can depend on n). This improves upon the aforementioned recent algorithm of Bansal et al. [9] which, for a similar approximation ratio, runs in time $\exp(n/r^r)n^{O(1)}$. For the other two problems, our algorithms give $O(r)$-approximation in the same running time, which improves upon a known $O(r)$-approximation algorithms with running time $\exp(n/2^r)n^{O(1)}$ that follow from [17]:

▶ **Theorem 1** (Main Theorem). *For any $r > 1$, there are $\exp(n/2^{r^2})n^{O(1)}$-time $(2 - 1/O(r))$-approximation algorithm for Vertex Cover on n-vertex graphs, and, $\exp(n/2^{r^2})n^{O(1)}$-time $O(r)$-approximation algorithms for Uniform Sparsest Cut and Balanced Separator.*

We remark that, when we take $r = C\sqrt{\log n}$ for a sufficiently large constant C, our algorithms coincide with the best polynomial time algorithms known for all three problems [32, 6].

1.2 Other Related Works

To prove Theorem 1, we use the Sum-of-Square relaxations of the problems and employ the conditioning framework from [13, 52] together with the main structural lemma from Arora, Rao and Vazirani's work [6]. We will describe how these parts fit together in Section 2. Before we do so, let us briefly discuss some related works not yet mentioned.

Sum-of-Square Relaxation and the Conditioning Framework. The Sum-of-Square (SoS) algorithm [47, 49, 41] is a generic yet powerful meta-algorithm that can be utilized to any polynomial optimization problems. The approach has found numerous applications in both continuous and combinatorial optimization problems. Most relevant to our work is the conditioning framework developed in [13, 52]. Barak et al. [13] used it to provide an algorithm for Unique Games with similar guarantee to [2], while Raghavendra and Tan [52] used the technique to give improved approximation algorithms for CSPs with cardinality constraints. A high-level overview of this framework is given in Sections 2.2 and 2.3.

Approximability of Vertex Cover, Sparsest Cut and Balanced Separator. All three problems studied in our work are very well studied in the field of approximation algorithms and hardness of approximation. For Vertex Cover, the greedy 2-approximation algorithm has been known since the 70's (see e.g. [25]). Better $(2 - \Omega(\frac{\log\log n}{\log n}))$-approximation algorithms were independently discovered in [11] and [45]. These were finally improved by Karakostas [32] who used the ARV Structural Theorem to provide a $(2 - \Omega(1/\sqrt{\log n}))$-approximation for the problem. On the lower bound side, Hastad [29] show that $(7/6 - \varepsilon)$-approximation for Vertex Cover is NP-hard. The ratio was improved in [21] to 1.36. The line of works that very recently obtained the proof of the (imperfect) 2-to-1 game conjecture [36, 19, 20, 37] also yield NP-hardness of $(\sqrt{2} - \varepsilon)$-approximate Vertex Cover as a byproduct. On the other hand, the Unique Games Conjecture (UGC) [33] implies that approximating Vertex Cover to within a factor $(2 - \varepsilon)$ is NP-hard [38, 10]. We remark here that only Hastad reduction (together with Moshkovitz-Raz PCP [46]) implies an almost exponential lower bond in terms of the running time, assuming ETH. Putting it differently, it could be the case that Vertex Cover can be approximated to within a factor 1.5 in time say $2^{O(\sqrt{n})}$, without refuting any complexity conjectures or hypotheses mentioned here. Indeed, the question of whether a subexponential time $(2 - \varepsilon)$-approximation algorithm for Vertex Cover exists for some constant $\varepsilon > 0$ was listed as an "interesting" open question in [2], and it remains so even after our work.

As for (Uniform) Sparsest Cut and Balanced Separator, they were both studied by Leighton and Rao who gave $O(\log n)$-approximation algorithms for the problems [43]. The ratio was improved in [6] to $O(\sqrt{\log n})$. In terms of hardness of approximation, these problems are *not* known to be NP-hard or even UGC-hard to approximate to even just 1.001 factor. (In contrast, the non-uniform versions of both problems are hard to approximate under UGC [18, 39].) Fortunately, inapproximability results of Sparsest Cut and Balanced Separator are known under stronger assumptions [22, 34, 51]. Specifically, Raghavendra et al. [51] shows that both problems are hard to approximate to any constant factor under the Small Set Expansion Hypothesis (SSEH) [50]. While it is not known whether SSEH follows from UGC, they are similar in many aspects, and indeed subexponential time algorithms for Unique Games [2, 13] also work for the Small Set Expansion problem. This means, for example, that there could be an $O(1)$-approximation algorithm for both problems in subexponential time without contradicting with any of the conjectures. Whether such algorithm exists remains an intriguing open question.

Fast Exponential Time Approximation Algorithms. As mentioned earlier, Bansal et al. [9] recently gave a "better than brute force" approximation algorithm for Vertex Cover. Their technique is to first observe that we can use branch-and-bound on the high-degree vertices; once only the low-degree vertices are left, they use Halperin's (polynomial time) approximation algorithm for Vertex Cover on bounded degree graphs [28] to obtain a good approximation. This approach is totally different than ours, and, given that the only way known to obtain $(2 - \Omega(1/\sqrt{\log n}))$-approximation in polynomial time is via the ARV Theorem, it is unlikely that their approach can be improved to achieve similar trade-off as ours.

[9] is not the first work that gives exponential time approximation algorithms for Vertex Cover. Prior to their work, Bourgeois et al. [15] gives a $(2-1/O(r))$-approximation $\exp(n/r)$-time algorithm for Vertex Cover; this is indeed a certain variant of the "limited brute force" algorithm. Furthermore, Bansal et al. [9] remarked in their manuscript that Williams and Yu have also independently come up with algorithms with similar guarantees to theirs, but, to the best of our knowledge, Williams and Yu's work is not yet made publicly available.

For Sparsest Cut and Balanced Separator, it is possible to derive $O(r)$-approximation algorithms that run in $\exp(n/2^r)$-time from a work of Charikar et al. [17]. In particular, it was shown in [17] that, for any metric space of n elements, if every subset of $n/2^r$ elements can be embedded isometrically into ℓ_1, then the whole space can be embedded into ℓ_1 with distortion $O(r)$. Since d-level of Sherali-Adams relaxations for both problems ensure that every d-size subset of the corresponding distance metric space can be embedded isometrically into ℓ_1, $(n/2^r)$-level of Sherali-Adams relaxations, which can be solved in $\exp(n/2^{\Omega(r)})$ time, give $O(r)$-approximation for both problems.

Organization. In the next section, we describe the overview of our algorithms. Then, in Section 3, we formalize the notations and state some preliminaries. The main lemma regarding conditioned SoS solution and its structure is stated in Section 4 (and proved in the appendix). This lemma is subsequently used in all our algorithms which are summarized in Section 5. We conclude our paper with several open questions in Section 6.

2 Overview of Technique

Our algorithms follow the "conditioning" framework developed in [13, 52]. In fact, our algorithms are very simple provided the tools from this line of work, and the ARV structural theorem from [6, 42]. To describe the ideas behind our algorithm, we will first briefly explain the ARV structural theorem and how conditioning works with Sum-of-Squares hierarchy in the next two subsections. Then, in the final subsection of this section, we describe the main insight behind our algorithms. For the ease of explaining the main ideas, we will sometimes be informal in this section; all algorithms and proofs will be formalized in the sequel.

Due to space constraint, we will focus only on the c-Balanced Separator problem; the algorithms for the two other problems can be found in the full version. In the c-Balanced Separator problem, we are given a graph $G = (V, E)$ and the goal is to find a partition of V into S_0 and $S_1 = V \setminus S_0$ that minimizes the number of edges across the cut (S_0, S_1) while also ensuring that $|S_0|, |S_1| \geqslant c'n$ for some constant $c' \in (0, c)$ where $n = |V|$. Note that the approximation ratio is the ratio between the number of edges cut by the solution and the optimal under the condition $|S_0|, |S_1| \geqslant cn$. (That is, this is a pseudo approximation rather than a true approximation; this is also the notion used in previous works [43, 6].) For the purpose of exposition, we focus only on the case where $c = 1/3$.

2.1 The ARV Structural Theorem

The geometric relaxation used in [6] embeds each vertex $i \in V$ into a point $v_i \in \mathbb{R}^d$ such that $\|v_i\|_2 = 1$. For a partition (S_0, S_1), the intended solution is $v_i = v_\emptyset$ if $i \in S_0$ and $v_i = -v_\emptyset$ otherwise, where $v_\emptyset$ is some unit vector. As a result, the objective function here is $\sum_{(i,j)\in E} \frac{1}{4}\|v_i - v_j\|_2^2$, and the cardinality condition $|S_0|, |S_1| \geqslant n/3$ is enforced by $\sum_{i,j\in V} \|v_i - v_j\|_2^2 \geqslant 8n/9$. Furthermore, Arora et al. [6] also employ the triangle inequality: $\|v_i - v_j\|_2^2 \leqslant \|v_i - v_k\|_2^2 + \|v_k - v_j\|_2^2$ for all $i, j, k \in V$. In other words, this relaxation can be written as follows.

$$\text{minimize} \quad \sum_{(i,j)\in E} \frac{1}{4}\|v_i - v_j\|_2^2 \tag{1}$$

$$\text{subject to} \quad \sum_{i,j\in[n]} \|v_i - v_j\|_2^2 \geqslant 8n/9 \tag{2}$$

$$\|v_i\|_2^2 = 1 \qquad \forall i \in V \tag{3}$$

$$\|v_i - v_j\|_2^2 \leqslant \|v_i - v_k\|_2^2 + \|v_k - v_j\|_2^2 \qquad \forall i, j, k \in V \tag{4}$$

Note here that the above relaxation can be phrased as a semidefinite program and hence can be solved to arbitrarily accuracy in polynomial time. The key insight shown by Arora et al. is that, given a solution $\{v_i\}_{i \in V}$ to the above problem, one can find two sets of vertices T, T' that are $\Omega(1/\sqrt{\log n})$ apart from each other, as stated below. Note that this version is in fact from [42]; the original theorem of [6] has a worst parameter with $\Delta = \Omega((\log n)^{2/3})$.

▶ **Theorem 2** (ARV Structural Theorem [6, 42]). *Let $\{v_i\}_{i \in V}$ be any vectors in $\mathbb{R}^d$ satisfying (2), (3), (4). There exist $T, T' \subseteq V$ each of size $\Omega(n)$ such that, for every $i \in T$ and $j \in T'$, $\|v_i - v_j\|_2^2 \geqslant \Delta = \Omega(1/\sqrt{\log n})$. Moreover, such sets can be found in randomized polynomial time.*

Note that, given the above theorem, it is easy to arrive at the $\Omega(1/\sqrt{\log n})$-approximation algorithm for balanced separator. In particular, we can pick a number θ uniformly at random from $[0, \Delta)$ and then output $S_0 = \{i \in V \mid \exists j \in T, \|v_i - v_j\|_2^2 \leqslant \theta\}$ and $S_1 = V \setminus S_0$. It is easy to check that the probability that each edge $(i, j) \in E$ is cut is at most $\|v_i - v_j\|_2^2/\Delta = O(\sqrt{\log n} \cdot \|v_i - v_j\|_2^2)$. Moreover, we have $|S_0| \geqslant |T| \geqslant \Omega(n)$ and $|S_1| \geqslant |T'| \geqslant \Omega(n)$, meaning that we have arrived at an $O(\sqrt{\log n})$-approximate solution for Balanced Separator.

An interesting aspect of the proof of [42] is that the bound on Δ can be improved if the solution $\{v_i\}_{i \in V}$ is "hollow" in the following sense: for every $i \in V$, the ball of radius[1] 0.1 around i contains few other vectors v_j's. In particular, if there are only m such v_j's, then Δ can be made $\Omega(1/\sqrt{\log m})$, instead of $\Omega(1/\sqrt{\log n})$ in the above version. We will indeed use this more fine-grained version in our algorithms. To the best of our knowledge, this version of the theorem has not yet been used in other applications of the ARV Theorem.

▶ **Theorem 3** (Refined ARV Structural Theorem [6, 42]). *Let $\{v_i\}_{i \in V}$ be any vectors in $\mathbb{R}^d$ satisfying (2), (3), (4) and let $m = \max_{i \in V} |\{j \in V \mid \|v_i - v_j\|_2^2 \leqslant 0.01\}|$. There exist $T, T' \subseteq V$ each of size $\Omega(n)$ such that, for every $i \in T$ and $j \in T'$, $\|v_i - v_j\|_2^2 \geqslant \Delta = \Omega(1/\sqrt{\log m})$. Moreover, such sets can be found in randomized polynomial time.*

2.2 Conditioning in Sum-of-Square Hierarchies

Another crucial tool used in our algorithm is Sum-of-Square hierarchy and the conditioning technique developed in [13, 52]. Perhaps the most natural interpretation of the sum-of-square solution with respect to the conditioning operation is to view the solution as local distributions. One can think of a degree-d sum-of-square solution for Balanced Separator as a collection of local distributions μ_S over $\{0,1\}^S$ for subsets of vertices $S \subseteq V$ of sizes at most d that satisfies certain consistency and positive semi-definiteness conditions, and additional linear constraints corresponding to $|S_0|, |S_1| \geqslant n/3$ and the triangle inequalities. More specifically, for every $U \subseteq V$ and every $\phi : U \to \{0,1\}$, the degree-d sum-of-squares solution gives us $\Pr_{\mu_U}[\forall j \in U, j \in S_{\phi(j)}]$ which is a number between zero and one. The consistency constraints ensure that these distributions are locally consistent; that is, for every $U' \subseteq U \in \{0,1\}$, the marginal distribution of μ_U on U' is equal to $\mu_{U'}$. We remark here that, for Balanced Separator and other problems considered in this work, a solution to the degree-d SoS relaxation for them can be found in time $\binom{n}{d}^{O(1)} = O(n/d)^{O(d)}$.

The consistency constraints on these local distributions allow us to define conditioning on local distributions in the same ways as typical conditional distributions. For instance, we can condition on the event $i \in S_0$ if $\Pr_{\mu_i}[i \in S_0] \neq 0$; this results in local distributions

[1] Here 0.1 can be changed to arbitrary positive constant.

$\{\mu'_U\}_{U \subseteq V, |U| \leqslant d-1}$ where μ'_U is the conditional distribution of $\mu_{U \cup \{i\}}$ on the event $i \in S_0$. In other words, for all $\phi : U \to \{0,1\}$,

$$\Pr_{\mu'_U}[\forall j \in U, j \in S_{\phi(j)}] = \frac{\Pr_{\mu_{U \cup \{i\}}}\left[i \in S_0 \wedge \left(\forall j \in U, j \in S_{\phi(j)}\right)\right]}{\Pr_{\mu_i}[i \in S_0]}.$$

Notice that the local distributions are now on subsets of at most $d-1$ vertices instead d vertices as before. That is, the conditioned solution is a degree-$(d-1)$ solution.

As for the semi-definiteness constraint, it suffices for the purpose of this discussion to think about only the degree-2 solution case. For this case, the semi-definiteness constraint in fact yields unit vectors $v_\emptyset, \{v_j\}_{j \in V}$ such that

$$\Pr_{\mu_i}[i \in S_0] = \frac{1 + \langle v_\emptyset, v_i \rangle}{2} \qquad \forall i \in V,$$

$$\Pr_{\mu_{\{i,j\}}}[i, j \in S_0] = \frac{1 + \langle v_\emptyset, v_i \rangle + \langle v_\emptyset, v_j \rangle + \langle v_i, v_j \rangle}{4} \qquad \forall i, j \in V.$$

It is useful to also note that the probability that i, j are on different side of the cut is exactly equal to $\frac{1}{4}\|v_i - v_j\|_2^2$; this is just because

$$\begin{aligned} \Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j] &= \Pr_{\mu_i}[i \in S_0] + \Pr_{\mu_j}[j \in S_0] - 2 \Pr_{\mu_{\{i,j\}}}[i \in S_0 \wedge j \in S_0] \\ &= \frac{1 - \langle v_i, v_j \rangle}{2} = \frac{1}{4}\|v_i - v_j\|_2^2, \end{aligned} \tag{5}$$

where Y_i, Y_j are boolean random variables such that $i \in S_{Y_i}$ and $j \in S_{Y_j}$.

Finally, note that the constraints for $|S_0|, |S_1| \geqslant n/3$ and the triangle inequalities are those that, when written in vector forms, translate to (2) and (4) from the ARV relaxation.

2.3 Our Algorithms: Combining Conditioning and ARV Theorem

The conditioning framework initiated in [13, 52] (and subsequently used in [8, 54, 44]) typically proceeds as follows: solve for a solution to a degree-d Sum-of-Square relaxation of the problem for a carefully chosen value of d, use (less than d) conditionings to make a solution into an "easy-to-round" degree-$O(1)$ solution, and finally round such a solution.

To try to apply this with the Balanced Separator problem, we first have to understand what are the "easy-to-round" solutions for the ARV relaxation. In this regards, first observe that, due to the more refined version of the ARV Theorem (Theorem 3), the approximation ratio is actually $O(\sqrt{\log m})$ which can be much better than $O(\sqrt{\log n})$. In particular, if $m \leqslant 2^{O(r^2)}$, this already yields the desired $O(r)$-approximation algorithm. This will be one of the "easy-to-round" situations. Observe also that we can in fact relax the requirement even further: it suffices if $|\{j \in V \mid \|v_i - v_j\|_2^2 \leqslant 0.01\}| \leqslant m$ holds for a constant fraction of vertices $i \in V$. This is because we can apply Theorem 3 on only the set of such i's which would still result in well-separated set of size $\Omega(n)$. Recall also that from (5) the condition $\|v_i - v_j\|_2^2 \leqslant 0.01$ is equivalent to $\Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j] \leqslant 0.04$.

Another type of easy-to-round situation is when, for most (i.e. $0.9n$) of $i \in V$, $\Pr_{\mu_i}[i \in S_0] \notin [0.2, 0.8]$. In this case, $T = \{i \in V \mid \Pr_{\mu_i}[i \in S_0] < 0.2\}$ and $T' = \{j \in V \mid \Pr_{\mu_j}[j \in S_0] > 0.8\}$ is a pair of large well-separated sets; it is not hard to argue that both T, T' are at least $\Omega(n)$ and that, for every $i \in T$ and $j \in T'$, $\|v_i - v_j\|_2^2$ is at least 0.6.

To recap, it suffices for us to condition degree-d solution so that we end up in one of the following two "easy-to-round" cases to get an $O(r)$-approximation for the problem.

1. For at least $n/100$ vertices $i \in V$, we have $|\{j \in V \mid \Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j] \leqslant 0.04\}| \leqslant 2^{O(r^2)}$.
2. For at least $9n/10$ vertices $i \in V$, we have $\Pr_{\mu_i}[i \in S_0] \notin [0.2, 0.8]$.

Here we will pick our d to be $n/2^{r^2}$; the running time needed to solve for such a solution is indeed $O(n/d)^{O(d)} = \exp(n/2^{\Omega(r^2)})n^{O(1)}$ as claimed. Now, suppose that we have a degree-d solution that does not belong to any of the two easy-to-round cases as stated above. This means that there must be $i \in V$ such that $\Pr_{\mu_i}[i \in S_0] \notin [0.2, 0.8]$ and that $|\{j \in V \mid \Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j] \leqslant 0.04\}| > 2^{O(r^2)}$. For simplicity, let us also assume for now that $\Pr_{\mu_i}[i \in S_0] = 0.5$. We will condition on the event $i \in S_0$; let the local distributions after conditioning be $\{\mu'_U\}_{U \subseteq V, |U| \leqslant d-1}$. Consider each $j \in V$ such that $\Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j] \leqslant 0.04$. Observe first that, before the conditioning, we have $\Pr_{\mu_j}[j \in S_0] \geqslant \Pr_{\mu_i}[i \in S_0] - \Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j] > 0.4$ and $\Pr_{\mu_j}[j \in S_0] \geqslant \Pr_{\mu_i}[i \in S_0] + \Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j] < 0.6$.

On the other hand, after the conditioning, we have

$$\Pr_{\mu'_j}[j \in S_0] = \frac{\Pr_{\mu_{\{i,j\}}}[i \in S_0, j \in S_0]}{\Pr_{\mu_i}[i \in S_0]} \geqslant \frac{\Pr_{\mu_i}[i \in S_0] - \Pr_{\mu_{\{i,j\}}}[Y_i \neq Y_j]}{\Pr_{\mu_i}[i \in S_0]} > 0.9.$$

Thus, this conditioning makes at least 2^{r^2} vertices j's such that $\Pr_{\mu_j}[j \in S_0] \in [0.2, 0.8]$ beforehand satisfy $\Pr_{\mu'_j}[j \in S_0] \notin [0.2, 0.8]$ afterwards. If we ignore how conditioning affects the remaining variables for now, this means that, after $n/2^{r^2}$ such conditioning all vertices $j \in V$ must have $\Pr_{\mu_j}[j \in S_0] \in [0.2, 0.8]$. Hence, we have arrived at an "easy-to-round" solution and we are done! The effect to the other variables that we ignored can easily be taken into account via a simple potential function argument and by considering conditioning on both $i \in S_0$ and $i \in S_1$; this part of the argument can be found in Section 4. This concludes the overview of our algorithm.

3 Preliminaries

3.1 Sum-of-Square Hierarchy and Conditioning

We define several notations regarding the Sum-of-Square (SoS) Hierarchy; these notations are based mainly on [12, 48]. We will only state preliminaries necessary for our algorithms. We recommend interested readers to refer to [48, 14] for a more thorough survey on SoS.

Let $\mathbb{R}_d[X_1, \dots, X_n]$ denote the set of all polynomials on $X_1, \dots, X_n$ of total degree at most d. First, we define *pseudo-expectation*, which represents solutions to SoS Hierarchy:

▶ **Definition 4** (Pseudo-Expectation). A *degree-d pseudo-expectation (p.e.)* is a linear operator $\tilde{\mathbb{E}} : \mathbb{R}_d[X_1, \dots, X_n] \to \mathbb{R}$ that satisfies the following:

- (Normalization) $\tilde{\mathbb{E}}[1] = 1$.
- (Linearity) For any $p \in \mathbb{R}_d[X_1, \dots, X_n]$ and $q \in \mathbb{R}_d[X_1, \dots, X_n]$, $\tilde{\mathbb{E}}[p+q] = \tilde{\mathbb{E}}[p] + \tilde{\mathbb{E}}[q]$.
- (Positivity) For any $p \in \mathbb{R}_{\lfloor d/2 \rfloor}[X_1, \dots, X_n]$, $\tilde{\mathbb{E}}[p^2] \geqslant 0$.

Furthermore, $\tilde{\mathbb{E}}$ is said to be *boolean* if $\tilde{\mathbb{E}}[(X_i^2 - 1)p] = 0$ for all $p \in \mathbb{R}_{d-2}[X_1, \dots, X_n]$.

Observe that, while $\tilde{\mathbb{E}}$ is a function over infinite domain, $\tilde{\mathbb{E}}$ has a succinct representation: due to its linearity, it suffices to specify the values of all monomials of total degree at most d and there are $n^{O(d)}$ such monomials. Furthermore, for boolean $\tilde{\mathbb{E}}$, we can save even further since it suffices to specify only products of at most d different variables. There are only $O(n/d)^{O(d)}$ such terms. From now on, we will only consider boolean pseudo-expectations. Note also that we use X_i as ± 1 variables instead of $0, 1$ variable as used in the proof overview. (Specifically, in the language of the overview section, $\Pr_{\mu_i}[i \in S_0]$ is now equal to $\tilde{\mathbb{E}}[(1 - X_i)/2]$.)

▶ **Definition 5.** A *system of polynomial constraints* $(\mathcal{P}, \mathcal{Q})$ consists of a set of equality constraints $\mathcal{P} = \{p_i = 0\}_{i \in |\mathcal{P}|}$ and a set of inequality constraints $\mathcal{Q} = \{q_j \geqslant 0\}_{j \in |\mathcal{Q}|}$, where p_i, q_j are polynomials over $X_1, \dots, X_n$.

For every $S \subseteq [n]$, we use X_S to denote the monomial $\prod_{i \in S} X_i$. Furthermore, for every $S \subseteq [n]$ and every $\phi : S \to \{\pm 1\}$, let X_ϕ be the polynomial $\prod_{i \in S}(1 + \phi(i)X_i)$. A boolean degree-$d$ p.e. $\tilde{\mathbb{E}}$ is said to *satisfy* a system $(\mathcal{P}, \mathcal{Q})$ if the following conditions hold:

- For all $p \in \mathcal{P}$ and all $S \subseteq [n]$ such that $|S| \leqslant d - \deg(p)$, we have $\tilde{\mathbb{E}}[X_S p] = 0$.
- For all $q \in \mathcal{Q}$, all $S \subseteq [n]$ s.t. $|S| \leqslant d - \deg(q)$ and all $\phi : S \to \{\pm 1\}$, we have $\tilde{\mathbb{E}}[X_\phi q] \geqslant 0$.

It is not hard to verify that finding a degree-d p.e. satisfying $(\mathcal{P}, \mathcal{Q})$ can be written as a semidefinite program of size $O(n/d)^{O(d)} \cdot |\mathcal{P}| \cdot |\mathcal{Q}|$ and hence can be solved in $\text{poly}(O(n/d)^d \cdot |\mathcal{P}| \cdot |\mathcal{Q}|)$ time. Finally, we define conditioning in terms of pseudo-expectation:

▶ **Definition 6** (Conditioning)**.** Let $\tilde{\mathbb{E}} : \mathbb{R}_d[X_1, \dots, X_n] \to \mathbb{R}$ be any boolean degree-d pseudo-expectation for some $d > 2$. For any $b \in \{\pm 1\}$ such that $\tilde{\mathbb{E}}[X_i] \neq -b$, we denote the conditional pseudo-expectation of $\tilde{\mathbb{E}}$ on $X_i = b$ by $\tilde{\mathbb{E}}|_{X_i = b} : \mathbb{R}_{d-1}[X_1, \dots, X_n] \to \mathbb{R}$ where $\tilde{\mathbb{E}}|_{X_i=b}[p] = \frac{\tilde{\mathbb{E}}[p(1+bX_i)]}{\tilde{\mathbb{E}}[1+bX_i]}$ for all $p \in \mathbb{R}_{d-1}[X_1, \dots, X_n]$.

The proposition below is simple to check, using the identity $(1 + bX_i) = \frac{1}{2}(1 + bX_i)^2$.

▶ **Proposition 7.** *If $\tilde{\mathbb{E}}$ satisfies a system $(\mathcal{P}, \mathcal{Q})$, then $\tilde{\mathbb{E}}|_{X_i=b}$ also satisfies $(\mathcal{P}, \mathcal{Q})$.*

3.2 ARV Structural Theorems

Having defined appropriate notations for SoS, we now move on to another crucial preliminary: the ARV Structural Theorem. It will be useful to state the theorem in terms of both metrics and pseudo-expectation. Let us start by definitions of several notations for metrics.

▶ **Definition 8** (Metric-Related Notations)**.** A *metric* d on X is a function $d : X \times X \to \mathbb{R}_{\geqslant 0}$ that satisfies (1) $d(x, x) = 0$, (2) symmetry $d(x, y) = d(y, x)$ and (3) triangle inequality $d(x, z) \leqslant d(x, y) + d(y, z)$, for all $x, y, z \in X$. We use the following notations in this work:

- For $x \in X$ and $S, T \subseteq X$, $d(x, S) := \min_{y \in S} d(x, y)$ and $d(S, T) := \min_{y \in S} d(y, T)$.
- We say that S, T are Δ*-separated* iff $d(S, T) \geqslant \Delta$.
- The *diameter* of a metric space (X, d) denoted by $\text{diam}(X, d)$ is $\max_{x,y \in X} d(x, y)$.
- We say that (X, d) is α-spread if $\sum_{x,y \in X} d(x, y) \geqslant \alpha |X|^2$.
- An *(open) ball of radius r around x* denoted by $\mathcal{B}_d(x, r)$ is defined as $\{y \in X \mid d(x, y) < r\}$.
- A metric space (X, d) is said to be *$(r,\ m)$-hollow* if $|\mathcal{B}_d(x, r)| \leqslant m$ for all $x \in X$.

▶ **Definition 9** (Negative Type Metric)**.** A metric space (X, d) is said to be of *negative type* if $\sqrt{d}$ is Euclidean, i.e., for some $f : X \to \mathbb{R}^q$, $\|f(x) - f(y)\|_2^2 = d(x, y)$ for all $x, y \in X$.

The ARV Theorem can now be stated as follows:

▶ **Theorem 10** (ARV Structural Theorem - Metric Formulation [6, 42])**.** *Let $\alpha, r > 0$ be any positive real number and $m \in \mathbb{N}$ be any positive integer. For any negative type metric space (X, d) with diam$(d) \leqslant 1$ that is α-spread and (r, m)-hollow, there exist $\Omega_{\alpha,\beta}(1/\sqrt{\log m})$-separated subsets $T, T' \subseteq X$ each of size $\Omega(|X|)$. Moreover, these sets can be found in randomized polynomial time.*

We remark that the quantitative bound $\Delta = \Omega_{\alpha,\beta}(1/\sqrt{\log m})$ follows from Lee's version of the theorem [42] whereas the original version only have $\Delta = \Omega_{\alpha,\beta}(1/(\log m)^{2/3})$.

As we are using the ARV Theorem in conjunction with the SoS conditioning framework, it is useful to also state the theorem in SoS notations. To do so, let us first state the following fact, which can be easily seen via the fact that the moment matrix (with (i, j)-entry equal to $\tilde{\mathbb{E}}[X_iX_j]$) is positive semidefinite and thus is a Gram matrix for some set of vectors:

▶ **Proposition 11.** *Let $\tilde{\mathbb{E}} : \mathbb{R}_2[X_1, \dots, X_n] \to \mathbb{R}$ be any degree-2 p.e. that satisfies $\tilde{\mathbb{E}}[(X_i - X_j)^2] \leqslant \tilde{\mathbb{E}}[(X_i - X_k)^2] + \tilde{\mathbb{E}}[(X_k - X_j)^2]$ for all $i, j, k \in [n]$. Define $d_{\tilde{\mathbb{E}}} : [n] \times [n] \to \mathbb{R}_{\geqslant 0}$ by $d_{\tilde{\mathbb{E}}}(i, j) = \tilde{\mathbb{E}}[(X_i - X_j)^2]$. Then, $([n], d_{\tilde{\mathbb{E}}})$ is a negative type metric space.*

When it is clear which pseudo-expectation we are referring to, we may drop the subscript from $d_{\tilde{\mathbb{E}}}$ and simply write d. Further, we use all metric terminologies with $\tilde{\mathbb{E}}$ in the natural manner; for instance, we say that $S, T \subseteq [n]$ are Δ-separated if $d_{\tilde{\mathbb{E}}}(S, T) \geqslant \Delta$.

Theorem 10 can now be restated in pseudo-expectation notations as follows.

▶ **Theorem 12** (ARV Structural Theorem - SoS Formulation [6, 42])**.** *For any $\alpha, \beta > 0$ and $m \in \mathbb{N}$, let $\tilde{\mathbb{E}} : \mathbb{R}_2[X_1, \dots, X_n] \to \mathbb{R}$ be any degree-2 p.e. such that the following hold:*

- *(Boolean) For every $i \in [n]$, $\tilde{\mathbb{E}}[X_i^2] = 1$.*
- *(Triangle Inequality) For every $i, j, k \in [n]$, $\tilde{\mathbb{E}}[(X_i - X_j)^2] \leqslant \tilde{\mathbb{E}}[(X_i - X_k)^2] + \tilde{\mathbb{E}}[(X_k - X_j)^2]$.*
- *(Balance) $\sum_{i,j \in [n]} \tilde{\mathbb{E}}[(X_i - X_j)^2] \geqslant \alpha n^2$.*
- *(Hollowness) For all $i \in [n]$, $|\{j \in [n] \mid \tilde{\mathbb{E}}[X_iX_j] > 1 - \beta\}| \leqslant m$.*

Then, there exists a randomized polynomial time algorithm that, with probability 2/3, produces $\Omega_{\alpha,\beta}(1/\sqrt{\log m})$-separated subsets $T, T' \subseteq [n]$ each of size $\Omega_{\alpha,\beta}(n)$.

Notice that, for boolean $\tilde{\mathbb{E}}$, $\tilde{\mathbb{E}}[X_iX_j] = 1 - \tilde{\mathbb{E}}[(X_i - X_j)^2]/2 = 1 - d_{\tilde{\mathbb{E}}}(i, j)/2$. This means that $\{j \in [n] \mid \tilde{\mathbb{E}}[X_iX_j] > 1 - \beta\}$ is simply $\mathcal{B}_{d_{\tilde{\mathbb{E}}}}(i, 2\beta)$. Another point to notice is that the metric $d_{\tilde{\mathbb{E}}}$ can have $\mathrm{diam}(d_{\tilde{\mathbb{E}}})$ as large as 4, instead of 1 required in Theorem 10, but this poses no issue since we can scale all distances down by a factor of 4.

4 Conditioning Yields Easy-To-Round Solution

Our main tool is the following lemma on structure of conditioned solution:

▶ **Lemma 13.** *Let τ, γ be any positive real numbers such that $\tau^2 < \gamma < 1$. Given a boolean degree-d pseudo-expectation $\tilde{\mathbb{E}} : \mathbb{R}_d[X_1, \dots, X_n] \to \mathbb{R}$ for a system $(\mathcal{P}, \mathcal{Q})$ and an integer $\ell < d$, we can, in time $O(n/d)^{O(d)}$, find a boolean degree-$(d - \ell)$ pseudo-expectation $\tilde{\mathbb{E}}' : \mathbb{R}_{d-\ell}[X_1, \dots, X_n] \to \mathbb{R}$ for the system $(\mathcal{P}, \mathcal{Q})$ such that the following condition holds:*

- *Let $V_{(-\tau,\tau)} := \{i \in [n] \mid \tilde{\mathbb{E}}'[X_i] \in (-\tau, \tau)\}$ denote the set of indices of variables whose pseudo-expectation lies in $(-\tau, \tau)$ and, for each $i \in [n]$, let $C_\gamma(i) := \{j \in [n] \mid \tilde{\mathbb{E}}'[X_iX_j] \in [-\gamma, \gamma]\}$ denote the set of all indices j's such that $\tilde{\mathbb{E}}'[X_iX_j]$ lies in $[-\gamma, \gamma]$. Then, for all $i \in V_{(-\tau,\tau)}$, we have*
$$|V_{(-\tau,\tau)} \setminus C_\gamma(i)| \leqslant \frac{n}{\ell(\gamma - \tau^2)^2}.$$

In other words, the lemma says that, when d is sufficiently large, we can condition so that we arrive at a pseudo-expectation with the hollowness condition if we restrict ourselves to $V_{(-\tau,\tau)}$. Note here that, outside of $V_{(-\tau,\tau)}$, this hollowness condition does not necessarily hold. For instance, it could be that after conditioning all variables be come integral (i.e. $\tilde{\mathbb{E}}[X_i] \in \{\pm 1\}$). However, this is the second "easy-to-round" case for ARV theorem, so this does not pose a problem for us.

The proof of Lemma 13 is based on a potential function argument. In particular, the potential function we use is $\Phi(\tilde{\mathbb{E}}) = \sum_{i \in [n]} \tilde{\mathbb{E}}[X_i]^2$. The main idea is that, as long as there is a "bad" $i \in [n]$ that violates the condition in the lemma, we will be able to finding a conditioning that significantly increases Φ. However, Φ is always at most n, meaning that this cannot happens too many times and, thus, we must at some point arrive at a pseudo-expectation with no bad i. Due to space constraint, we defer the full proof to the appendix.

5 The Algorithms

All of our algorithms follow the same three-step blueprint, as summarized below.

Step I: Solving for Degree-$n/2^{\Omega(r^2)}$ Pseudo-Expectation. We first consider the system of constraints corresponding to the best known existing polynomial time algorithm for each problem, and we solve for degree-$n/2^{\Omega(r^2)}$ pseudo-expectation for such a system.

Step II: Conditioning to Get "Hollow" Solution. Then, we apply Lemma 13 to arrive at a degree-2 pseudo-expectation that satisfies the system and that additionally is hollow, i.e., $|V_{(-\tau,\tau)} \setminus C_\gamma(i)| \leqslant 2^{r^2}$ for all $i \in V_{(-\tau,\tau)}$ for appropriate values of τ, γ. Recall here that $V_{(-\tau,\tau)}$ and $C_\gamma(i)$ are defined in Lemma 13.

Step III: Following the Existing Algorithm. Finally, we follow the existing polynomial time approximation algorithms [6, 32, 1] to arrive at an approximate solution for the problem of interest. The improvement in the approximation ratio comes from the fact that our pseudo-expectation is now in the "easy-to-round" regime, i.e., the ARV Theorem gives separation of $\Omega(1/r)$ for this regime instead of $\Omega(1/\sqrt{\log n})$ for the general regime.

Let us now demonstrate our framework with the Balanced Separator problem.

▶ **Theorem 14.** *For any $r > 1$ (possibly depending on n), there exists an $\exp(n/2^{\Omega(r^2)})$ poly(n)-time $O(r)$-approximation for Balanced Separator on n-vertex graphs.*

Proof. On input graph $G = (V = [n], E)$, the algorithm works as follows.

Step I: Solving for Degree-$n/2^{\Omega(r^2)}$ Pseudo-Expectation. For every real number $OBJ \in \mathbb{R}$, let $(\mathcal{P}^{BS}_{G,OBJ}, \mathcal{Q}^{BS}_{G,OBJ})$ be the following system of equations:

1. (Boolean) For all $i \in [n]$, $X_i^2 - 1 = 0$.
2. (Balance) $\sum_{i,j\in[n]}(X_i - X_j)^2 - 16n^2/9 \geqslant 0, n/3 - \sum_{i\in[n]} X_i \geqslant 0$ and $\sum_{i\in[n]} X_i + n/3 \geqslant 0$.
3. (Triangle Inequalities) For all $i, j, k \in [n]$, $(X_i - X_k)^2 + (X_k - X_j)^2 - (X_i - X_j)^2 \geqslant 0, (1 - X_i)^2 + (1 - X_j)^2 - (X_i - X_j)^2 \geqslant 0$, and $(1 + X_i)^2 + (1 - X_j)^2 - (X_i - X_j)^2 \geqslant 0$.
4. (Objective Bound) $4 \cdot OBJ - \sum_{(i,j)\in E}(X_i - X_j)^2 \geqslant 0$.

Let $D := \lceil 1000n/2^{r^2} \rceil + 2$. The algorithm first uses binary search to find the largest OBJ such that there exists a degree-D pseudo-expectation for $(\mathcal{P}^{BS}_{G,OBJ}, \mathcal{Q}^{BS}_{G,OBJ})$. Let this value of OBJ be OBJ^*, and let $\tilde{\mathbb{E}}$ be a degree-D pseudo-expectation satisfying $(\mathcal{P}^{BS}_{G,OBJ^*}, \mathcal{Q}^{BS}_{G,OBJ^*})$.

Again, observe that this step takes $\exp\left(n/2^{\Omega(r^2)}\right)$ poly(n) time and $OBJ^* \leqslant OPT$ where OPT is the number of edges cut in the balanced separator of G.

Step II: Conditioning to Get "Hollow" Solution. Use Lemma 13 to find a degree-2 p.e. $\tilde{\mathbb{E}}'$ for $(\mathcal{P}^{BS}_{G,OBJ^*}, \mathcal{Q}^{BS}_{G,OBJ^*})$ such that for all $i \in V_{(-0.9,0.9)}$, $|V_{(-0.9,0.9)} \setminus C_{0.9}(i)| < 2^{r^2}$.

Step III: Following ARV Algorithm. The last step follows the ARV algorithm [6], which first uses the ARV lemma to obtain two large well separated set. While in the traditional setting, the structural theorem can be applied immediately; we have to be more careful and treat the two "easy-to-round" cases differently. This is formalized below.

▶ **Claim 15.** *There exist disjoint subsets $T, T' \subseteq [n]$ that are $\Omega(1/r)$-separated. Moreover, these subsets can be found (with probability 2/3) in polynomial time.*

Proof. For every $a, b \in \mathbb{R}$, let $V_{\geqslant a} = \{i \in [n] \mid \tilde{\mathbb{E}}'[X_i] \geqslant a\}$, $V_{\leqslant b} = \{i \in [n] \mid \tilde{\mathbb{E}}'[X_i] \leqslant b\}$ and $V_{(a,b)} = \{i \in [n] \mid a < \tilde{\mathbb{E}}'[X_i]| < b\}$. Let $\tau = 0.9$. We consider the following two cases:

1. $|V_{\geqslant\tau}| \geqslant 0.1n$ or $|V_{\leqslant-\tau}| \geqslant 0.1n$. Suppose without loss of generality that it is the former. We claim that $|V_{\leqslant 0.8}| \geqslant 0.2n$. To see that this is the case, observe that
$$n/3 = \sum_{i \in V \setminus V_{\leqslant 0.8}} \tilde{\mathbb{E}}'[X_i] + \sum_{i \in V_{\leqslant 0.8}} \tilde{\mathbb{E}}'[X_i] \quad \geqslant 0.8(n - |V_{\leqslant 0.8}|) - |V_{\leqslant 0.8}| = 0.8n - 1.8|V_{\leqslant 0.8}|$$
which implies that $|V_{\leqslant 0.8}| > 0.2n$. Let $T = V_{\geqslant\tau}$ and $T' = V_{\leqslant 0.8}$. As we have shown, $|T|, |T'| \geqslant \Omega(n)$. Moreover, for every $i \in T$ and $j \in T'$, triangle inequality implies that
$$\tilde{\mathbb{E}}'[X_iX_j] \leqslant \tilde{\mathbb{E}}'[X_iX_j] + \tilde{\mathbb{E}}'[(1 - X_i)(1 + X_j)] = 1 - \tilde{\mathbb{E}}'[X_i] + \tilde{\mathbb{E}}'[X_j] < 1 - 0.9 + 0.8 = 0.9.$$
That is, $\tilde{\mathbb{E}}'[(X_i - X_j)^2] = 2 - 2\tilde{\mathbb{E}}'[X_iX_j] > 0.2$, completing the proof for the first case.
2. $|V_{\geqslant\tau}| < 0.1n$ and $|V_{\leqslant-\tau}| < 0.1n$. This implies that $|V_{(-\tau,\tau)}| \geqslant 0.8n$. Moreover,
$$\sum_{i,j \in V_{(-\tau,\tau)}} \tilde{\mathbb{E}}[(X_i - X_j)^2] \geqslant 16n^2/9 - 8n|V_{\geqslant\tau}| - 8n|V_{\leqslant\tau}| > 0.1n^2.$$
Hence, applying the ARV Theorem (Theorem 12) to $V_{(-\tau,\tau)}$ yields the desired T, T'.

Thus, in both cases, we can find the desired T, T' in randomized polynomial time. ◀

Once we have found T, T', we use the following rounding scheme from [43, 6]: pick θ uniformly at random from $[0, d(T, T'))$. Then, output $(S, V \setminus S)$ where $S = \{i \in [n] \mid d(i, T) < \theta\}$.

Observe that $T \subseteq S$ and $T' \subseteq (V \setminus S)$, which means that $|S|, |V \setminus S| > \Omega(n)$; in other words, the output is a valid solution for the problem. Moreover, for every $(i, j) \in E$, it is easy to see that the probability that i and j end up in different sets is at most $|d(i, T) - d(j, T)|/d(T, T') \leqslant d(i, j)/d(T, T') \leqslant O(r) \cdot d(i, j)$. As a result, the expected number of edges cut is $O(r) \cdot \sum_{(i,j) \in E} d(i, j) = O(r) \cdot OBJ^*$, which completes our proof. ◀

6 Conclusion and Open Questions

In this work, we use the conditioning framework in the SoS Hierarchy together with the ARV Theorem to design "fast" exponential time approximation algorithms for Vertex Cover, Uniform Sparsest Cut and Balanced Separator that achieve significant speed-up over the trivial "limited brute force" algorithms. While we view this as a step towards understanding the time vs approximation ratio trade-off for these problems, many questions remain open.

First, as discussed in the introduction, current lower bounds do not rule out subexponential time approximation algorithms in the regime of our study. For instance, an 1.9-approximation algorithm for Vertex Cover could still possibly be achieved in say $2^{O(\sqrt{n})}$ time. Similarly for Uniform Sparsest Cut and Balanced Separator, $O(1)$-approximation for them could still possibly be achieved in subexponential time. The main open question is to either confirm that such algorithms exist, or rule them out under certain believable complexity hypotheses.

Another, perhaps more plausible, direction is to try to extend our technique to other problems for which the known polynomial time approximation algorithms employ the ARV Theorem. The most obvious candidates are Minimum UnCut, Minimum 2CNF Deletion, Directed Balanced Separator and Directed Sparsest Cut, which were shown in [1] to admit $O(\sqrt{\log n})$-approximation in polynomial time. While not yet rigorously verified, we believe that our technique also yields approximation algorithms with similar running time as Uniform Sparsest Cut to these problems as well. In this direction, there are also other potentially more challenging candidates, such as Balanced Vertex Separator, (Non-uniform) Sparsest Cut, and Minimum Linear Arrangement. While the first problem admits $O(\sqrt{\log n})$-approximation in polynomial time [23], additional ingredients beyond the ARV Theorem are required in the algorithm. On the other hand, the latter two problems only admit $O(\sqrt{\log n}\log\log n)$-approximation [4, 16, 24]. It seems challenging to remove this $\log\log n$ factor and achieve a constant factor approximation, even in our "fast" exponential time regime.

References

1. Amit Agarwal, Moses Charikar, Konstantin Makarychev, and Yury Makarychev. $O(\sqrt{\log n})$ approximation algorithms for min UnCut, min 2CNF deletion, and directed cut problems. In *STOC*, pages 573–581, 2005. `doi:10.1145/1060590.1060675`.
2. Sanjeev Arora, Boaz Barak, and David Steurer. Subexponential algorithms for unique games and related problems. *J. ACM*, 62(5):42:1–42:25, 2015. `doi:10.1145/2775105`.
3. Sanjeev Arora, Russell Impagliazzo, William Matthews, and David Steurer. Improved algorithms for unique games via divide and conquer. *ECCC*, 17:41, 2010.
4. Sanjeev Arora, James R. Lee, and Assaf Naor. Euclidean distortion and the sparsest cut. In *STOC*, pages 553–562, 2005. `doi:10.1145/1060590.1060673`.
5. Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998. `doi:10.1145/278298.278306`.
6. Sanjeev Arora, Satish Rao, and Umesh V. Vazirani. Expander flows, geometric embeddings and graph partitioning. *J. ACM*, 56(2):5:1–5:37, 2009. `doi:10.1145/1502793.1502794`.
7. Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: A new characterization of NP. *J. ACM*, 45(1):70–122, 1998. `doi:10.1145/273865.273901`.
8. Per Austrin, Siavosh Benabbas, and Konstantinos Georgiou. Better balance by being biased: A 0.8776-approximation for max bisection. In *SODA*, pages 277–294, 2013. `doi:10.1137/1.9781611973105.21`.
9. Nikhil Bansal, Parinya Chalermsook, Bundit Laekhanukit, Danupon Nanongkai, and Jesper Nederlof. New tools and connections for exponential-time approximation. *CoRR*, abs/1708.03515, 2017. `arXiv:1708.03515`.
10. Nikhil Bansal and Subhash Khot. Optimal long code test with one free bit. In *FOCS*, pages 453–462, 2009. `doi:10.1109/FOCS.2009.23`.
11. R. Bar-Yehuda and S. Even. A local-ratio theorem for approximating the weighted vertex cover problem. In G. Ausiello and M. Lucertini, editors, *Analysis and Design of Algorithms for Combinatorial Problems*, volume 109 of *North-Holland Mathematics Studies*, pages 27–45. North-Holland, 1985. `doi:10.1016/S0304-0208(08)73101-3`.
12. Boaz Barak, Fernando G. S. L. Brandão, Aram Wettroth Harrow, Jonathan A. Kelner, David Steurer, and Yuan Zhou. Hypercontractivity, sum-of-squares proofs, and their applications. In *STOC*, pages 307–326, 2012. `doi:10.1145/2213977.2214006`.
13. Boaz Barak, Prasad Raghavendra, and David Steurer. Rounding semidefinite programming hierarchies via global correlation. In *FOCS*, pages 472–481, 2011. `doi:10.1109/FOCS.2011.95`.
14. Boaz Barak and David Steurer. Sum-of-squares proofs and the quest toward optimal algorithms. *ECCC*, 21:59, 2014. URL: `http://eccc.hpi-web.de/report/2014/059`.
15. Nicolas Bourgeois, Bruno Escoffier, and Vangelis Th. Paschos. Approximation of max independent set, min vertex cover and related problems by moderately exponential algorithms.

Discrete Applied Mathematics, 159(17):1954–1970, 2011. doi:10.1016/j.dam.2011.07.009.

16 Moses Charikar, Mohammad Taghi Hajiaghayi, Howard J. Karloff, and Satish Rao. ℓ_2^2 spreading metrics for vertex ordering problems. *Algorithmica*, 56(4):577–604, 2010. doi:10.1007/s00453-008-9191-1.

17 Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Local global tradeoffs in metric embeddings. *SIAM J. Comput.*, 39(6):2487–2512, 2010. doi:10.1137/070712080.

18 Shuchi Chawla, Robert Krauthgamer, Ravi Kumar, Yuval Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. *Computational Complexity*, 15(2):94–114, 2006. doi:10.1007/s00037-006-0210-9.

19 Irit Dinur, Subhash Khot, Guy Kindler, Dor Minzer, and Muli Safra. Towards a proof of the 2-to-1 games conjecture? *ECCC*, 23:198, 2016. URL: http://eccc.hpi-web.de/report/2016/198.

20 Irit Dinur, Subhash Khot, Guy Kindler, Dor Minzer, and Muli Safra. On non-optimally expanding sets in grassmann graphs. *ECCC*, 24:94, 2017. URL: https://eccc.weizmann.ac.il/report/2017/094.

21 Irit Dinur and Shmuel Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162(1):439–485, 2005.

22 Uriel Feige. Relations between average case complexity and approximation complexity. In *STOC*, pages 534–543, 2002. doi:10.1145/509907.509985.

23 Uriel Feige, MohammadTaghi Hajiaghayi, and James R. Lee. Improved approximation algorithms for minimum weight vertex separators. *SIAM J. Comput.*, 38(2):629–657, 2008. doi:10.1137/05064299X.

24 Uriel Feige and James R. Lee. An improved approximation ratio for the minimum linear arrangement problem. *Inf. Process. Lett.*, 101(1):26–29, 2007. doi:10.1016/j.ipl.2006.07.009.

25 Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

26 Dima Grigoriev. Complexity of positivstellensatz proofs for the knapsack. *Computational Complexity*, 10(2):139–154, 2001. doi:10.1007/s00037-001-8192-0.

27 Venkatesan Guruswami and Ali Kemal Sinop. Lasserre hierarchy, higher eigenvalues, and approximation schemes for graph partitioning and quadratic integer programming with PSD objectives. In *FOCS*, pages 482–491, 2011. doi:10.1109/FOCS.2011.36.

28 Eran Halperin. Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs. *SIAM J. Comput.*, 31(5):1608–1623, 2002. doi:10.1137/S0097539700381097.

29 Johan Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 2001. doi:10.1145/502090.502098.

30 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.

31 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.

32 George Karakostas. A better approximation ratio for the vertex cover problem. *ACM Trans. Algorithms*, 5(4):41:1–41:8, 2009. doi:10.1145/1597036.1597045.

33 Subhash Khot. On the power of unique 2-prover 1-round games. In *CCC*, page 25, 2002. doi:10.1109/CCC.2002.1004334.

34 Subhash Khot. Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM J. Comput.*, 36(4):1025–1071, 2006. doi:10.1137/S0097539705447037.

35 Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable csps? *SIAM J. Comput.*, 37(1):319–357, 2007. doi:10.1137/S0097539705447372.

36 Subhash Khot, Dor Minzer, and Muli Safra. On independent sets, 2-to-2 games, and grassmann graphs. In *STOC*, pages 576–589, 2017. doi:10.1145/3055399.3055432.

37 Subhash Khot, Dor Minzer, and Muli Safra. Pseudorandom sets in grassmann graph have near-perfect expansion. *ECCC*, 25:6, 2018. URL: https://eccc.weizmann.ac.il/report/2018/006.

38 Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within 2-epsilon. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008. doi:10.1016/j.jcss.2007.06.019.

39 Subhash Khot and Nisheeth K. Vishnoi. The unique games conjecture, integrality gap for cut problems and embeddability of negative-type metrics into ℓ_1. *J. ACM*, 62(1):8:1–8:39, 2015. doi:10.1145/2629614.

40 Pravesh K. Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer. Sum of squares lower bounds for refuting any CSP. In *STOC*, pages 132–145, 2017.

41 Jean B. Lasserre. An explicit equivalent positive semidefinite program for nonlinear 0-1 programs. *SIAM Journal on Optimization*, 12(3):756–769, 2002. doi:10.1137/S1052623400380079.

42 James R. Lee. On distance scales, embeddings, and efficient relaxations of the cut cone. In *SODA*, pages 92–101, 2005. URL: http://dl.acm.org/citation.cfm?id=1070432.1070446.

43 Frank Thomson Leighton and Satish Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *J. ACM*, 46(6):787–832, 1999. doi:10.1145/331524.331526.

44 Pasin Manurangsi and Prasad Raghavendra. A birthday repetition theorem and complexity of approximating dense CSPs. *CoRR*, abs/1607.02986, 2016. URL: http://arxiv.org/abs/1607.02986, arXiv:1607.02986.

45 Burkhard Monien and Ewald Speckenmeyer. Ramsey numbers and an approximation algorithm for the vertex cover problem. *Acta Inf.*, 22(1):115–123, 1985. doi:10.1007/BF00290149.

46 Dana Moshkovitz and Ran Raz. Two-query PCP with subconstant error. *J. ACM*, 57(5):29:1–29:29, 2010. doi:10.1145/1754399.1754402.

47 Yurii Nesterov. Squared functional systems and optimization problems. In *High performance optimization*, pages 405–440. Springer, 2000.

48 Ryan O'Donnell and Yuan Zhou. Approximability and proof complexity. In *SODA*, pages 1537–1556, 2013. doi:10.1137/1.9781611973105.111.

49 Pablo A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization.* PhD thesis, California Institute of Technology, 2000.

50 Prasad Raghavendra and David Steurer. Graph expansion and the unique games conjecture. In *STOC*, pages 755–764, 2010. doi:10.1145/1806689.1806788.

51 Prasad Raghavendra, David Steurer, and Madhur Tulsiani. Reductions between expansion problems. In *CCC*, pages 64–73, 2012. doi:10.1109/CCC.2012.43.

52 Prasad Raghavendra and Ning Tan. Approximating CSPs with global cardinality constraints using SDP hierarchies. In *SODA*, pages 373–387, 2012. URL: http://portal.acm.org/citation.cfm?id=2095149&CFID=63838676&CFTOKEN=79617016.

53 Grant Schoenebeck. Linear level lasserre lower bounds for certain k-csps. In *FOCS*, pages 593–602, 2008. doi:10.1109/FOCS.2008.74.

54 Yuichi Yoshida and Yuan Zhou. Approximation schemes via sherali-adams hierarchy for dense constraint satisfaction problems and assignment problems. In *ITCS*, pages 423–438, 2014. doi:10.1145/2554797.2554836.

A Proof of the Conditioning Lemma

In this section, we prove Lemma 13. To facilitate our proof, let us prove a simple identity regarding the potential change for a single variable after conditioning:

▶ **Proposition 16.** *Let $\tilde{\mathbb{E}} : \mathbb{R}_d[X_1, \dots, X_n] \to \mathbb{R}$ be any degree-d pseudo-expectation for some $d > 2$ and let $i \in [n]$ be such that $\tilde{\mathbb{E}}[X_i] \neq -1, 1$. Then, for any $j \in [n]$, we have*

$$\left(\frac{1-\tilde{\mathbb{E}}[X_i]}{2}\right)\left(\tilde{\mathbb{E}}|_{X_i=-1}[X_j]\right)^2 + \left(\frac{1+\tilde{\mathbb{E}}[X_i]}{2}\right)\left(\tilde{\mathbb{E}}|_{X_i=1}[X_j]\right)^2 - \tilde{\mathbb{E}}[X_j]^2 = \frac{(\tilde{\mathbb{E}}[X_iX_j] - \tilde{\mathbb{E}}[X_i]\tilde{\mathbb{E}}[X_j])^2}{1-\tilde{\mathbb{E}}[X_i]^2}.$$

Proof. For succinctness, let $a = (1-\tilde{\mathbb{E}}[X_i])/2$, $b = \tilde{\mathbb{E}}|_{X_i=1}[X_j]$ and $c = \tilde{\mathbb{E}}|_{X_i=-1}[X_j]$. Observe that, from definition of conditioning, we have

$$ab + (1-a)c = \tilde{\mathbb{E}}[(1-X_i)X_j/2] + \tilde{\mathbb{E}}[(1+X_i)X_j/2] = \tilde{\mathbb{E}}[X_j].$$

Hence, the left hand side term of the equation in the proposition statement can be rewritten as

$$ab^2 + (1-a)c^2 - (ab + (1-a)c)^2 = a(1-a)(b-c)^2. \tag{6}$$

Let $\mu_i = \tilde{\mathbb{E}}[X_i]$. Now, observe that $b - c$ is simply

$$\begin{aligned}
\tilde{\mathbb{E}}|_{X_i=-1}[X_j] - \tilde{\mathbb{E}}|_{X_i=1}[X_j] &= \frac{\tilde{\mathbb{E}}[(1-X_i)X_j]}{1-\mu_i} - \frac{\tilde{\mathbb{E}}[(1+X_i)X_j]}{1+\mu_i} \\
&= \frac{\tilde{\mathbb{E}}[(1+\mu_i)(1-X_i)X_j - (1-\mu_i)(1+X_i)X_j]}{1-\mu_i^2} \\
&= \frac{\tilde{\mathbb{E}}[2(\mu_i - X_i)X_j]}{1-\mu_i^2} \\
&= \frac{2(\mu_i\tilde{\mathbb{E}}[X_j] - \tilde{\mathbb{E}}[X_iX_j])}{1-\mu_i^2}.
\end{aligned}$$

Plugging the above equality back into (6) yields the desired identity. ◀

With the above lemma ready, we now proceed to the proof of Lemma 13. Before we do so, let us also note that our choice of potential function $\tilde{\mathbb{E}}[X_i]^2$ is not of particular importance; indeed, there are many other potential functions that work, such as the entropy of X_i.

Proof of Lemma 13. We describe an algorithm below that finds $\tilde{\mathbb{E}}'$ by iteratively conditioning the pseudo-distribution on the variable X_i that violates the condition.

1. Let $\tilde{\mathbb{E}}_0 = \tilde{\mathbb{E}}$
2. For $t = 1, \dots, \ell$, execute the following steps.
   a. Let $V^{t-1}_{(-\tau,\tau)} := \{i \in [n] \mid \tilde{\mathbb{E}}_{t-1}[X_i] \in (-\tau, \tau)\}$.
      Moreover, for each $i \in [n]$, let $C^{t-1}_{\gamma}(i) := \{j \in [n] \mid \tilde{\mathbb{E}}_{t-1}[X_iX_j] \in [-\gamma, \gamma]\}$.
   b. If $|V^{t-1}_{(-\tau,\tau)} \setminus C^{t-1}_{\gamma}(i)| \leqslant \frac{n}{\ell(\gamma-\tau^2)^2}$ for all $i \in V^{t-1}_{(-\tau,\tau)}$, then output $\tilde{\mathbb{E}}_{t-1}$ and terminate.
   c. Otherwise, pick $i \in V^{t-1}_{(-\tau,\tau)}$ such that $|V^{t-1}_{(-\tau,\tau)} \setminus C^{t-1}_{\gamma}(i)| > \frac{n}{\ell(\gamma-\tau^2)^2}$. Compute $\Phi(\tilde{\mathbb{E}}_t|_{X_i=1})$ and $\Phi(\tilde{\mathbb{E}}_t|_{X_i=-1})$ and let $\tilde{\mathbb{E}}_t$ be equal to the one with larger potential.
3. If the algorithm has not terminated, output NULL.

Notice that, if the algorithm terminates in Step 2b, then the output pseudo-distribution obviously satisfies the condition in Lemma 13. Hence, we only need to show that the algorithm always terminates in Step 2b (and never reaches Step 3). Recall that we let $\Phi(\tilde{\mathbb{E}})$ denote $\sum_{i\in[n]} \tilde{\mathbb{E}}[X_i]^2$. To prove this, we will analyze the change in $\Phi(\tilde{\mathbb{E}}_t)$ over time. In particular, we can show the following:

▶ **Claim 17.** *For every $t \in [\ell]$, $\Phi(\tilde{\mathbb{E}}_t) - \Phi(\tilde{\mathbb{E}}_{t-1}) > n/\ell$.*

Proof. First, notice that it suffices to prove the following because $\tilde{\mathbb{E}}_{t-1}[(1-X_i)/2]+\tilde{\mathbb{E}}_{t-1}[(1+X_i)/2] = 1$ and, from our choice of $\tilde{\mathbb{E}}_t$, we have $\Phi(\tilde{\mathbb{E}}_t) = \max\{\Phi(\tilde{\mathbb{E}}_{t-1}|_{X_i=1}), \Phi(\tilde{\mathbb{E}}_{t-1}|_{X_i=-1})\}$.

$$\left(\tilde{\mathbb{E}}\left[\frac{1-X_i}{2}\right] \cdot \Phi(\tilde{\mathbb{E}}_{t-1}|_{X_i=-1}) + \tilde{\mathbb{E}}\left[\frac{1+X_i}{2}\right] \cdot \Phi(\tilde{\mathbb{E}}_{t+1}|_{X_i=1})\right) - \Phi(\tilde{\mathbb{E}}_{t-1}) > n/\ell.$$

Recall that, from our definition of Φ, the left hand side above can simply be written as

$$\sum_{j\in[n]} \left(\tilde{\mathbb{E}}\left[\frac{1-X_i}{2}\right]\left(\tilde{\mathbb{E}}_{t-1}|_{X_i=-1}[X_j]\right)^2 + \tilde{\mathbb{E}}\left[\frac{1+X_i}{2}\right]\left(\tilde{\mathbb{E}}_{t-1}|_{X_i=1}[X_j]\right)^2 - \tilde{\mathbb{E}}_{t-1}[X_j]^2\right).$$

From Proposition 16, this is equal to

$$\begin{aligned}\sum_{j\in[n]} \frac{\left(\tilde{\mathbb{E}}_{t-1}[X_iX_j] - \tilde{\mathbb{E}}_{t-1}[X_i]\tilde{\mathbb{E}}_{t-1}[X_j]\right)^2}{1-\tilde{\mathbb{E}}_{t-1}[X_i]^2} &\geqslant \sum_{j\in V^{t-1}_{(-\tau,\tau)}\setminus C^{t-1}_\gamma(i)} \frac{\left(\tilde{\mathbb{E}}_{t-1}[X_iX_j] - \tilde{\mathbb{E}}_{t-1}[X_i]\tilde{\mathbb{E}}_{t-1}[X_j]\right)^2}{1-\tilde{\mathbb{E}}_{t-1}[X_i]^2} \\ &> \sum_{j\in V^{t-1}_{(-\tau,\tau)}\setminus C^{t-1}_\gamma(i)} \left(\gamma-\tau^2\right)^2 \\ \left(\text{From } |V^{t-1}_{(-\tau,\tau)} \setminus C^{t-1}_\gamma(i)| \geqslant \frac{n}{\ell(\gamma-\tau^2)^2}\right) &> n/\ell,\end{aligned}$$

where the second inequality follows from $|\tilde{\mathbb{E}}[X_i]|, |\tilde{\mathbb{E}}[X_j]| < \tau$ and $|\tilde{\mathbb{E}}[X_iX_j]| > \gamma$ for all $j \in V^{t-1}_{(-\tau,\tau)} \setminus C^{t-1}_\gamma(i)$. ◀

It is now easy to see that Claim 17 implies that the algorithm never reaches Step 3. Otherwise, we would have $\Phi(\tilde{\mathbb{E}}_\ell) > n/\ell + \Phi(\tilde{\mathbb{E}}_{\ell-1}) > \cdots > n + \Phi(\tilde{\mathbb{E}}) > n$, a contradiction. ◀