Approximately Counting Locally-Optimal Structures[☆]

Leslie Ann Goldberg[a], Rob Gysel[b], John Lapinskas[a,1]

[a]*Department of Computer Science, University of Oxford, Parks Road, OX1 3QD, UK.*
[b]*Department of Computer Science, University of California, 2063 Kemper Hall, One Shields Avenue, Davis, CA 95616-8562, US.*

Abstract

In general, constructing a locally-optimal structure is a little harder than constructing an arbitrary structure, but significantly easier than constructing a globally-optimal structure. A similar situation arises in listing. In counting, most problems are #P-complete, but in approximate counting we observe an interesting reversal of the pattern. Assuming that #BIS is not equivalent to #SAT under AP-reductions, we show that counting maximal independent sets in bipartite graphs is harder than counting maximum independent sets. Motivated by this, we show that various counting problems involving minimal separators are #SAT-hard to approximate. These problems have applications for constructing triangulations and phylogenetic trees.

1. Introduction

A *locally-optimal* structure is a combinatorial structure that cannot be improved by certain (greedy) local moves, even though it may not be globally optimal. An example is a maximal independent set in a graph. It is trivial to construct an independent set in a graph (for example, the singleton set containing any vertex is an independent set). It is easy to construct a maximal independent set (the greedy algorithm can do this). However, it is NP-hard to construct a globally-optimal independent set, which in this case means a maximum independent set. In the setting in which we work, this situation is typical. Constructing a locally-optimal structure is somewhat more difficult than constructing an arbitrary structure, and constructing a globally-optimal structure

[☆]The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007–2013) ERC grant agreement no. 334828. The paper reflects only the authors' views and not the views of the ERC or the European Commission. The European Union is not liable for any use that may be made of the information contained therein. The research leading to these results has also received funding from the National Science Foundation, grant number IIS-1219278. A preliminary version [16] of this paper appears in the proceedings of ICALP 2015.

Email addresses: `leslie.goldberg@cs.ox.ac.uk` (Leslie Ann Goldberg), `rsgysel@ucdavis.edu` (Rob Gysel), `john.lapinskas@cs.ox.ac.uk` (John Lapinskas)

[1]Corresponding author.

is more difficult than constructing a locally-optimal structure. For example, in bipartite graphs, it is trivial to construct an independent set, easy to (greedily) construct a maximal independent set, and more difficult to construct a maximum independent set (even though this can be done in polynomial time). This general phenomenon has been well-studied. In 1987, Johnson, Papadimitriou and Yannakakis [22] defined the complexity class PLS (for "polynomial-time local search") that captures local optimisation problems where one iteration of the local search algorithm takes polynomial time. As the authors point out, practically all empirical evidence leads to the conclusion that finding locally-optimal solutions is much easier than solving NP-hard problems, and this is supported by complexity-theoretic evidence, since a problem in PLS cannot be NP-hard unless NP=co-NP. An example that illustrates this point is the graph partitioning problem. For this problem it is trivial to find a valid partition, and it is NP-hard to find a globally-optimal (minimum weight) partition but Schäffer and Yannakakis [27] showed that finding a locally-optimal solution (with respect to a particular swapping-dynamics) is PLS-complete, so is presumably of intermediate complexity.

For listing combinatorial structures, a similar pattern emerges. By self-reducibility, there is a nearly-trivial polynomial-space polynomial-delay algorithm for listing the independent sets of a graph [15]. A polynomial-space polynomial-delay algorithm for listing the *maximal* independent sets exists, due to Tsukiyama et al. [31], but it is more complicated. On the other hand, there is no polynomial-space polynomial-delay algorithm for listing the *maximum* independent sets unless P=NP. There is a polynomial-space polynomial-delay algorithm for listing the maximum independent sets of a bipartite graph [23], but this is substantially more complicated than any of the previous algorithms.

When we move from constructing and listing to counting, these differences become obscured because nearly everything is #P-complete. For example, counting independent sets, maximal independent sets, and maximum independent sets of a graph are all #P-complete problems, even if the graph is bipartite [32]. Furthermore, even *approximately* counting independent sets, maximal independent sets, and maximum independent sets of a graph are all #P-complete with respect to approximation-preserving reductions [10].

The purpose of this paper is to highlight an interesting situation that arises in approximate counting where, contrary to the situations that we have just discussed, approximately counting locally-optimal structures is apparently more difficult than counting globally-optimal structures.

In order to explain the result, we first briefly summarise what is known about the complexity of approximate counting within #P. This will be explained in more detail in Section 2. There are three relevant complexity classes — the class containing problems which admit a fully-polynomial randomised approximation scheme (FPRAS), the class $\#\mathrm{RH}\Pi_1$, and #P itself. Dyer et al. [10] showed that #BIS, the problem of counting independent sets in a bipartite graph, is complete for $\#\mathrm{RH}\Pi_1$ with respect to approximation-preserving (AP) reductions and that #IS, the problem of counting independent sets in a (general) graph is #P-complete with respect to AP-reductions. It is generally believed that the

$\#\text{RH}\Pi_1$-complete problems are not FPRASable, but that they are of intermediate complexity, and are not as difficult to approximate as the problems which are #P-complete with respect to AP-reductions. Many problems have subsequently been shown to be $\#\text{RH}\Pi_1$-complete and #P-complete with respect to AP-reductions. More examples will be given in Section 2.

We can now describe the interesting situation which emerges with respect to independent sets in bipartite graphs. Dyer et al. [10] showed that approximately counting independent sets and approximately counting *maximum* independent sets are both $\#\text{RH}\Pi_1$-complete with respect to AP-reductions. Thus, the pattern outlined above would suggest that approximately counting *maximal* independent sets in bipartite graphs ought to also be $\#\text{RH}\Pi_1$-complete. However, we show (Theorem 1, below) that approximately counting *maximal* independent sets in bipartite graphs is actually #P-complete with respect to AP-reductions. Thus, either $\#\text{RH}\Pi_1$ and #P are equivalent in approximation complexity (contrary to the picture that has been emerging in earlier papers), or this is a scenario where approximately counting locally-optimal structures is actually more difficult than approximately counting globally-optimal ones.

Motivated by the difficulty of approximately counting maximal independent sets in bipartite graphs, we also study the problem of approximately counting other locally-optimal structures that arise in algorithmic applications. First, the problem of counting the *minimal separators* of a graph arises in diverse applications from triangulation theory to phylogeny construction in computational biology. A minimal separator is a particular type of vertex separator. Definitions are given in Section 1.1. Algorithmic applications arise because fixed-parameter-tractable algorithms are known whose running time is polynomial in the number of minimal separators of a graph. These algorithms were originally developed by Bouchitté and Todinca [5, 6] (and improved in [11]) to exactly solve the so-called *treewidth* and *minimum-fill* problems. The former problem, finding the exact treewidth of a graph, is widely studied due to its applicability to a number of other NP-complete problems [4]. The technique has recently been generalized [14] to cover problems including *treecost* [2] and *treelength* [26]. The algorithm can also be used to find a minimum-width *tree-decomposition* of a graph, a key data structure that is used to solve a variety of NP-complete problems in polynomial time when the width of the tree-decomposition is fixed [4]. In recent years, much research has been dedicated to exact-exponential algorithms for treewidth [3], the fastest of which [12] has running time closely connected to the number of minimal separators in the graph. Indeed, there exist polynomials p_L and p_U such that if the graph has n vertices and M minimal separators, then the running time is at least $p_L(n)M$ and at most $p_U(n)M^2$.

Bouchitté and Todinca's approach has also recently been applied to solve the *perfect phylogeny problem* and two of its variants [21]. In this problem, the input is a set of phylogenetic characters, each of which may be viewed as a partition of a subset of *species*. The goal is to find a phylogenetic tree such that every character is *convex* on that tree — that is, the parts of each partition form connected subtrees that do not overlap. Such a tree is called a *perfect phylogeny*.

In all of these applications, it would be useful to count the minimal vertex separators of a graph, since this would give an a priori bound on the running time of the algorithms. Thus, we consider the difficulty of this problem, whose complexity was previously unresolved, even in terms of exact computation. Theorem 2 shows that the problem of counting minimal separators is #P-complete, both with respect to Turing reductions (for exact computation) and with respect to AP-reductions. Thus, this problem is as difficult to approximate as any problem in #P.

Motivated by applications to treewidth [11] and phylogeny [20, 21], we also consider various heuristic approximations to the minimal separator problem. The number of inclusion-minimal separators is a natural choice for a lower bound on the number of minimal separators. Conversely, the number of (s, t)-minimal separators, taken over all vertices s and t, is a natural choice for an upper bound on the number of minimal separators. Theorem 2 shows that both of these bounds are difficult to compute, either exactly or approximately. Finally, the number and structure of 2-component minimal separators is important in computational biology. 2-component minimal separators arise naturally in the problem of determining whether a subset of "quartet phylogenies" can be assembled uniquely [20]. Thus, we study the problem of counting such minimal separators. Theorem 2 shows that they are complete for #P with respect to exact and approximate computation.

Our new results about counting minimal vertex separators are obtained by first considering the problem of counting minimal edge separators. These locally-optimal structures are also known as *bonds* or *minimal cuts*, and are well-studied in other contexts — see e.g. Diestel [9]. Theorem 3 gives the first hardness result for counting these structures, either exactly or approximately.

In addition to studying maximal independent sets and minimal vertex and edge separators, we study two other locally-optimal structures related to maximal independent sets in bipartite graphs. A maximal independent set is precisely an independent set in a graph which is also a *dominating set*. Theorem 4 shows that counting dominating sets in bipartite graphs is #P-hard with respect to AP-reductions. It is already known to be #P-hard to compute exactly [24]. Finally, in Theorems 5 and 6 we show that maximal independent sets in bipartite graphs can be represented as unions of sets, so a set union problem #SetUnions is also #P-hard with respect to AP-reductions, and so is its inverse #UnionReps.

1.1. Detailed Results

We now give formal definitions of the problems that we study, and state our results precisely. Note that all problems are indexed for reference at the end of the paper. Our first result is that counting maximal independent sets in a bipartite graph is #P-complete with respect to AP-reductions (even though counting maximum independent sets in bipartite graphs is only $\#\mathrm{RH}\Pi_1$-complete with respect to these reductions). For readers that are unfamiliar with AP-reductions, details are given in Section 2.

Definition 1. *Let G be a graph. We say that an independent set $X \subseteq V(G)$ of G is* maximal *if no proper superset of X is an independent set of G.*

Problem 1. #MaximalBIS.
Input: *A bipartite graph G.*
Output: *The number of maximal independent sets of G.*

The following theorem is proved in Section 3.

Theorem 1. #MaximalBIS $\equiv_{\mathrm{AP}}$ #SAT.

Next we state our results relating to counting minimal separators. In the following definitions, $G = (V, E)$ is a graph, s and t are distinct vertices of G, and $X \subseteq V$ is a set of vertices.

Definition 2. *X is an (s, t)*-separator *of G if s and t lie in different components of $G - X$. If, in addition, no proper subset of X is an (s, t)-separator of G, then we say that X is a* minimal (s, t)-separator *of G.*

Definition 3. *X is a* minimal separator *of G if it is a minimal (s, t)-separator for some $s, t \in V$.*

For example, let $G = (V, E)$ be the graph defined by

$$V = \{1, 2, 3, 4, 5\}, \quad \text{and} \quad E = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 1\}, \{1, 5\}\}.$$

G is a four-edge cycle with a pendant vertex. Then $\{1, 3\}$ is a minimal separator of G since it is a minimal $(2, 4)$-separator.

We have already seen that algorithms for counting and approximately counting minimal separators are useful in algorithmic applications. There is also lots of existing work on listing minimal separators. Given a graph G, let n be the number of vertices and let m be the number of edges. Kloks and Kratsch, and independently, Sheng and Liang, showed how to compute all (s, t)-minimal separators in $O(n^3)$ time per (s, t)-minimal separator [25, 28]. Computing all minimal separators by computing (s, t)-minimal separators for each possible vertex pair in this way leads to an $O(n^5)$ time per minimal separator listing algorithm. Berry, Bordat, and Cogis [1] improved this approach, computing all minimal separators in $O(n^3)$ time per minimal separator. Each of these algorithms require storing minimal separators in an adequate data structure. Takata's algorithm [30] generates the set of minimal separators in $O(n^3 m)$ time per minimal separator but linear space. A graph has at most $O(1.6181^n)$ minimal separators [13]. We study the following computational problems, based on our desire to count and to approximately count minimal separators.

Problem 2. #(s, t)-BiMinimalSeps.
Input: *A bipartite graph G and two vertices $s, t \in V(G)$.*
Output: *The number of minimal (s, t)-separators of G, which we denote by* $\mathrm{MS}(G, s, t)$.

Problem 3. #BiMinimalSeps.
Input: *A bipartite graph G.*
Output: *The number of minimal separators of G, which we denote by* $\mathrm{MS}(G)$.

Theorem 2 below shows that both problems are #P-complete to solve exactly and are complete for #P with respect to approximation-preserving reductions.

Motivated by applications to phylogeny [20] we also consider various heuristic approximations to the minimal separator problem. We start by defining the notion of an inclusion-minimal separator, since the number of these is a natural lower bound for the number of minimal separators.

Definition 4. *Let G be a graph. A minimal separator X of G is said to be an* inclusion-minimal separator *if no proper subset of X is a minimal separator.*

In the five-vertex example above, the minimal separator $\{1, 3\}$ is not an inclusion-minimal separator since $\{1\} \subset \{1, 3\}$ is a minimal $(5, 4)$-separator. However $\{1\}$ is an inclusion-minimal separator. We consider the following computational problem.

Problem 4. #BiInclusionMinimalSeps.
Input: *A bipartite graph G.*
Output: *The number of inclusion-minimal separators of G, which we denote by* $\mathrm{IMS}(G)$.

We also consider the problem of counting 2-component minimal separators since these arise in phylogenetic assembly.

Problem 5. #(s, t)-BiConnMinimalSeps.
Input: *A bipartite graph G and two vertices $s, t \in V(G)$.*
Output: *The number of minimal (s, t)-separators X of G such that $G - X$ has exactly two connected components.*

Problem 6. #BiConnMinimalSeps.
Input: *A bipartite graph G.*
Output: *The number of minimal separators X of G such that $G - X$ has exactly two connected components.*

Finally, our main theorem about minimal separators shows that all of these problems are #P-complete and are also complete for #P with respect to AP-reductions.

Theorem 2. *The problems* #(s, t)-BiMinimalSeps, #BiMinimalSeps, #(s, t)-BiConnMinimalSeps, #BiConnMinimalSeps *and* #BiInclusionMinimalSeps *are* #P*-complete and are equivalent to* #SAT *under AP-reduction.*

Theorem 2 is proved in Section 4. In order to prove it, we first study algorithmic problems related to other natural locally-optimal structures, namely minimal edge-separators. These problems are interesting for their own sake, but they are also used in the proof of Theorem 2. In the following definitions, $G = (V, E)$ is again a graph, and s and t are distinct vertices of G. $F \subseteq E$ is a set of edges of G.

Definition 5. *F is an* (s,t)-edge separator *of G if s and t lie in different components of $G - F$. If in addition no proper subset of F is an (s,t)-edge separator of G then we say that F is a* minimal (s,t)-edge separator *of G.*

Definition 6. *F is a* minimal edge separator *of G if it is a minimal (s,t)-edge separator for some $s,t \in V$.*

As the following proposition shows, there is no need to define inclusion-minimal edge separators, since these would be the same as minimal edge separators.

Proposition 7. *Let $G = (V,E)$ be a connected graph. An edge separator $F \subseteq E$ of G is minimal if and only if no proper subset of F is an edge separator of G.*

Proof. This is immediate from a slightly more general proposition, Proposition 12, which in turn is a result of Whitney [33]. $\square$

We study the following problems, showing that they are both #P-complete with respect to AP-reductions and #P-complete to compute exactly.

Problem 7. #(s,t)-BiMinimalEdgeSeps.
Input: *A bipartite graph G and two vertices $s,t \in V(G)$.*
Output: *The number of minimal (s,t)-edge separators of G, which we denote by* $\mathrm{MES}(G,s,t)$.

Problem 8. #BiMinimalEdgeSeps.
Input: *A bipartite graph G.*
Output: *The number of minimal edge separators of G, which we denote by* $\mathrm{MES}(G)$.

Theorem 3. *The problems* #BiMinimalEdgeSeps *and* #(s,t)-BiMinimalEdgeSeps *are* #P*-complete and are equivalent to* #SAT *under AP-reduction.*

In addition to studying maximal independent sets and minimal vertex and edge separators, we study two other structures related to maximal independent sets in bipartite graphs.

Definition 8. *Let G be a graph. We say that a set $X \subseteq V(G)$ is a* dominating set *in G if every vertex in $V(G) \setminus X$ sends an edge into X.*

We consider the following computational problem.

Problem 9. #BiDomSets.
Input: *A bipartite graph G.*
Output: *The number of dominating sets in G.*

It is already known [24] that exactly counting dominating sets in bipartite graphs is #P-complete. We show that that approximately counting them is also complete for #P with respect to AP-reductions.

Theorem 4. #BiDomSets $\equiv_{\mathrm{AP}}$ #SAT.

Finally, we show that maximal independent sets in bipartite graphs can be represented as unions of sets, so a natural set union problem is also #P-hard with respect to AP-reductions, and so is its inverse. To describe the problem, we use the following notation. Throughout the paper, we write $\mathbb{N}$ for the set $\{1, 2, \dots\}$ of natural numbers. For all $n \in \mathbb{N}$, we write $[n] = \{1, 2, \dots, n\}$.

Definition 9. *Let $\mathcal{F} \subseteq 2^{[n]}$. We define $\cup\mathcal{F} = \bigcup_{F \in \mathcal{F}} F$, $\mathcal{U}(\mathcal{F}) = \{\cup\mathcal{F}' \mid \mathcal{F}' \subseteq \mathcal{F}\}$, and $\mathcal{U}_{\mathcal{F}}^{-1}(F) = \{\mathcal{F}' \subseteq \mathcal{F} \mid \cup\mathcal{F}' = F\}$.*

For example, taking $\mathcal{F} = \{\{1\}, \{1, 2\}, \{3, 4\}\}$, we have

$$\begin{aligned}
\cup\mathcal{F} &= [4],\\
\mathcal{U}(\mathcal{F}) &= \{\{1\}, \{1, 2\}, \{3, 4\}, \{1, 3, 4\}, \{1, 2, 3, 4\}\},\\
\mathcal{U}_{\mathcal{F}}^{-1}([4]) &= \{\{\{1, 2\}, \{3, 4\}\}, \{\{1\}, \{1, 2\}, \{3, 4\}\}\}.
\end{aligned}$$

Note in particular that we may have $\mathcal{U}_{\mathcal{F}}^{-1}(F) = \emptyset$.

The following theorems are proved in Section 5.

Problem 10. #SetUnions.
Input: *An integer $n \in \mathbb{N}$ and a family of sets $\mathcal{F} \subseteq 2^{[n]}$.*
Output: $|\mathcal{U}(\mathcal{F})|$.

Theorem 5. #SetUnions $\equiv_{\mathrm{AP}}$ #SAT.

Note that the connection between the two problems driving Theorem 5 was already known in the context of the union-closed sets conjecture — see Bruhn, Charbit, Schaudt and Telle [7]. We give an explicit proof for clarity.

Problem 11. #UnionReps.
Input: *An integer $n \in \mathbb{N}$ and a family of sets $\mathcal{F} \subseteq 2^{[n]}$.*
Output: $|\mathcal{U}_{\mathcal{F}}^{-1}([n])|$.

Theorem 6. #UnionReps $\equiv_{\mathrm{AP}}$ #SAT.

2. Preliminaries

Let X and Y be sets. Then we write $X \subseteq Y$ if X is a subset of Y, and $X \subset Y$ if X is a proper subset of Y. We write 2^X for the power set of X. For $t \in \mathbb{N}$, we write $X^{(t)}$ for the set of subsets of X of cardinality t.

Let X and Y be multisets. We write $X \uplus Y$ for the disjoint union of X and Y. We also adopt the convention that elements of a multiset with the same name are nevertheless distinguishable.

We require our graphs to be simple, i.e. to have no loops or multiple edges. We require our multigraphs to have no loops. Let $G = (V, E)$ be a multigraph. For all $v \in V$, we write $N(v) = \{w \in V : \{v, w\} \in E\}$. For all $S \subseteq V$, we write $N(S) = \bigcup_{v \in S} N(v)$. We define the underlying graph of G to be the graph with vertex set V and edge set $\{e : e \in E\}$.

Let $G=(V,E)$ be a graph. If $F\subseteq E$, we write $G-F$ for the graph $(V,E\backslash F)$. If $X\subseteq V$, we write $G-X$ for the graph $G[V\setminus X]$ induced by G on $V\setminus X$.

The following definitions are standard in the field, and have been taken largely from [19]. We require our problem inputs to be given as finite binary strings, and write Σ^* for the set of all such strings. A *randomised approximation scheme* is an algorithm for approximately computing the value of a function $f:\Sigma^*\to\mathbb{N}$. The approximation scheme has a parameter $\varepsilon\in(0,1)$ which specifies the error tolerance. A *randomised approximation scheme* for f is a randomised algorithm that takes as input an instance $x\in\Sigma^*$ (e.g. an encoding of the graph G in an instance of #MaximalBIS) and a rational error tolerance $\varepsilon\in(0,1)$, and outputs a rational number z (a random variable depending on the "coin tosses" made by the algorithm) such that, for every instance x, $\mathbb{P}(e^{-\varepsilon}f(x)\le z\le e^{\varepsilon}f(x))\ge\frac{3}{4}$. The randomised approximation scheme is said to be a *fully polynomial randomised approximation scheme*, or *FPRAS*, if it runs in time bounded by a polynomial in $|x|$ and ε^{-1}.

Our main tool for understanding the relative difficulty of approximation counting problems is *approximation-preserving reductions*. We use the notion of AP-reduction from Dyer et al. [10]. Suppose that f and g are functions from Σ^* to $\mathbb{N}$. An AP-reduction from f to g gives a way to turn an FPRAS for g into an FPRAS for f. An *approximation-preserving reduction* or *AP-reduction* from f to g is a randomised algorithm $\mathcal{A}$ for computing f using an oracle for g. The algorithm $\mathcal{A}$ takes as input a pair $(x,\varepsilon)\in\Sigma^*\times(0,1)$, and satisfies the following three conditions: (i) every oracle call made by $\mathcal{A}$ is of the form (w,δ), where $w\in\Sigma^*$ is an instance of g, and $\delta\in(0,1)$ is an error bound satisfying $\delta^{-1}\le\mathrm{poly}(|x|,\varepsilon^{-1})$; (ii) the algorithm $\mathcal{A}$ meets the specification for being a randomised approximation scheme for f (as described above) whenever the oracle meets the specification for being a randomised approximation scheme for g; and (iii) the run-time of $\mathcal{A}$ is polynomial in $|x|$ and ε^{-1} and the bit-size of the values returned by the oracle.

If an AP-reduction from f to g exists we write $f\le_{\mathrm{AP}}g$, and say that *f is AP-reducible to g*. Note that if $f\le_{\mathrm{AP}}g$ and g has an FPRAS then f has an FPRAS. (The definition of AP-reduction was chosen to make this true.) If $f\le_{\mathrm{AP}}g$ and $g\le_{\mathrm{AP}}f$ then we say that *f and g are equivalent under AP-reduction*, and write $f\equiv_{\mathrm{AP}}g$. A word of warning about terminology: the notation $\le_{\mathrm{AP}}$ has been used (see e.g. [8]) to denote a different type of approximation-preserving reduction which applies to optimisation problems. We will not study optimisation problems in this paper, so hopefully this will not cause confusion.

Dyer et al. [10] studied counting problems in #P and identified three classes of counting problems that are interreducible under AP-reductions. The first class, containing the problems that have an FPRAS, are trivially equivalent under AP-reduction since all the work can be embedded into the reduction (which declines to use the oracle). The second class is the set of problems that are equivalent to #SAT, the problem of counting satisfying assignments to a Boolean formula in CNF, under AP-reduction. These problems are complete for #P with respect to AP-reductions. Zuckerman [34] has shown that #SAT cannot have an FPRAS unless RP = NP. The same is obviously true of any problem to which #SAT is AP-reducible.

The third class appears to be of intermediate complexity. It contains all of the counting problems expressible in a certain logically-defined complexity class, $\#\mathrm{RH}\Pi_1$. Typical complete problems include counting the downsets in a partially ordered set [10], computing the partition function of the ferromagnetic Ising model with local external magnetic fields [17], and counting the independent sets in a bipartite graph, which is formally defined as follows.

Problem 12. #BIS.
Input: *A bipartite graph G.*
Output: *The number of independent sets in G, which we denote by* $\mathrm{IS}(G)$.

In [10] it was shown that #BIS is complete for the logically-defined complexity class $\#\mathrm{RH}\Pi_1$ with respect to AP-reductions. Goldberg and Jerrum [18] have conjectured that there is no FPRAS for #BIS. Early indications point to the fact that it may be of intermediate complexity, between the FPRASable problems and those that are complete for #P with respect to AP-reductions.

3. Hardness of #MaximalBIS

We first prove that #MaximalBIS is complete for #P with respect to AP-reductions. We reduce from the well-known problem of counting independent sets in an arbitrary graph.

Problem 13. #IS.
Input: *A graph G.*
Output: *The number of independent sets in G.*

Note that #IS is complete for #P with respect to AP-reductions — indeed, the following appears as Theorem 3 of Dyer, Goldberg, Greenhill and Jerrum [10].

Theorem 7. *(DGGJ)* $\#\mathrm{IS} \equiv_{\mathrm{AP}} \#\mathrm{SAT}$.

We can now prove Theorem 1.

Theorem 1. $\#\mathrm{MaximalBIS} \equiv_{\mathrm{AP}} \#\mathrm{SAT}$.

Proof. Since #MaximalBIS is in #P, $\#\mathrm{MaximalBIS} \leq_{\mathrm{AP}} \#\mathrm{SAT}$ follows from [10]. To go the other direction, we will show $\#\mathrm{IS} \leq_{\mathrm{AP}} \#\mathrm{MaximalBIS}$. Let $\mathrm{MIS}(G)$ denote the number of maximal independent sets in a graph G. Let $G = (V, E)$ be an instance of #IS. Without loss of generality let $V = [n]$ for some $n \in \mathbb{N}$, let $m = |E|$, and let $t = n + 2$. We shall construct an instance G' of #MaximalBIS with the property that $\mathrm{IS}(G) \leq \mathrm{MIS}(G')/2^{tm} \leq \mathrm{IS}(G) + \frac{1}{4}$, which will be sufficient for the reduction. See Figure 1 for an example.

Informally, we obtain a bipartite graph G' (an instance of #MaximalBIS) from G by first t-thickening and then 4-stretching each of G's edges and by also

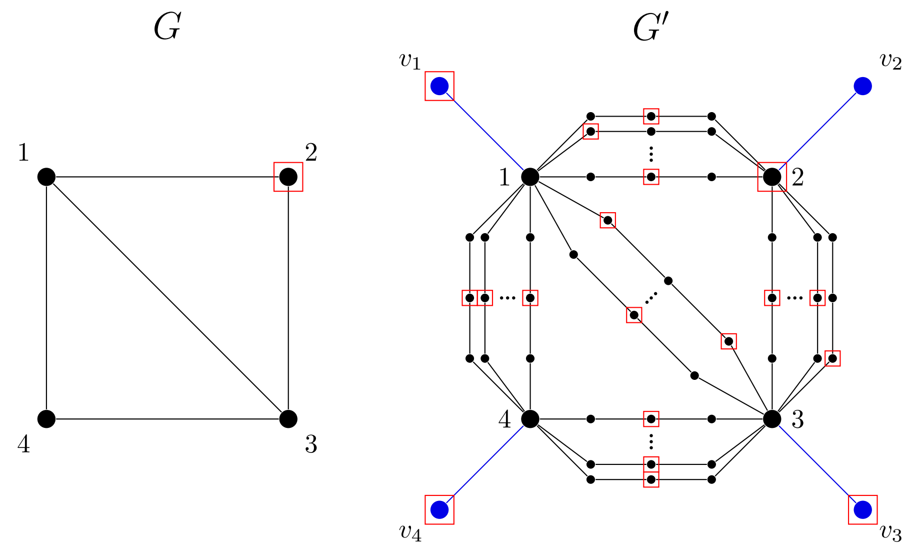

Figure 1: An example of the reduction from an instance G of #IS to an instance G' of #MaximalBIS used in the proof of Theorem 1. The boxes around vertices indicate a non-maximal independent set in G and one of its maximal counterparts in G'. Note in particular how the presence of v_4 ensures that vertex 4 has an occupied neighbour in G'.

adding a bristle to each of G's vertices. Formally, we define G' as follows. For each $e \in E$ let X_e, Y_e and Z_e be sets of t vertices. We require all of these sets to be disjoint from each other and from $[n]$. Write $X_e = \{x_e^k \mid k \in [t]\}$, $Y_e = \{y_e^k \mid k \in [t]\}$, and $Z_e = \{z_e^k \mid k \in [t]\}$. Also, let $W = \bigcup_{e \in E} X_e \cup Y_e \cup Z_e$. Let $V^* = \{v_1, \ldots, v_n\}$ be a set of distinct vertices which is disjoint from $[n] \cup W$. Then we define

$$\begin{aligned} V(G') &= [n] \cup V^* \cup W, \\ E(G') &= \{\{i, v_i\} \mid i \in [n]\} \cup \bigcup_{\substack{e=\{i,j\}\in E \\ i<j \\ k\in[t]}} \{\{i, x_e^k\}, \{x_e^k, y_e^k\}, \{y_e^k, z_e^k\}, \{z_e^k, j\}\}. \end{aligned}$$

Let $S \subseteq [n]$ be an arbitrary set. We shall determine the number $\mathrm{MIS}_S(G')$ of maximal independent sets $T \subseteq V(G')$ with $T \cap [n] = S$, and thereby bound $\mathrm{MIS}(G')$.

First, note that for every $S \subseteq [n]$, the set $S \cup \{v_i \in V^* \mid i \notin S\} \cup \bigcup_e Y_e$ is a maximal independent set of G', so $\mathrm{MIS}_S(G')$ is non-zero. Also, if T is a maximal independent set of G' and $T \cap [n] = S$ then $T \cap V^* = \{v_i \in V^* \mid i \notin S\}$. In particular, this implies that every unoccupied vertex in $[n]$ has an occupied neighbour in V^*.

Consider an edge $e = \{i, j\} \in E$, where $i < j$, and a value $k \in [t]$. If T is a maximal independent set of G' containing both i and j then $T \cap \{x_e^k, y_e^k, z_e^k\} =$

$\{y_e^k\}$. On the other hand, if T is a maximal independent set of G' containing i but not j then $T \cap \{x_e^k, y_e^k, z_e^k\}$ can either be $\{y_e^k\}$ or $\{z_e^k\}$. This choice can be made independently for each $k \in [t]$. Similarly, if T is a maximal independent set of G' containing neither of i and j then $T \cap \{x_e^k, y_e^k, z_e^k\}$ can either be $\{x_e^k, z_e^k\}$, or $\{y_e^k\}$.

Given $S \subseteq [n]$, let $\mu(S)$ be the number of edges of G with both endpoints in S. We conclude from the previous observations that $\mathrm{MIS}_S(G') = 2^{(m-\mu(S))t}$ so $\mathrm{MIS}(G') = \sum_{S \subseteq [n]} 2^{(m-\mu(S))t}$. Since each independent set S of G has $\mu(S) = 0$, $\mathrm{MIS}(G') \geq \mathrm{IS}(G) 2^{mt}$. Furthermore, since there are at most 2^n sets $S \subseteq [n]$ that are not independent sets of G, and each of these has $\mu(S) \geq 1$, we have

$$\mathrm{IS}(G) \leq \frac{\mathrm{MIS}(G')}{2^{tm}} \leq \mathrm{IS}(G) + 2^n 2^{-t} = \mathrm{IS}(G) + \frac{1}{4}. \tag{1}$$

Equation (1) implies that there is an AP-reduction from #IS to #MaximalBIS. The details of the reduction showing how to tune the accuracy parameter in the oracle call for approximating $\mathrm{MIS}(G')$ in order to get a sufficiently good approximation to $\mathrm{IS}(G)$ are exactly as in the proof of Theorem 3 of [10].

□

4. Minimal separator problems

4.1. Two intermediate problems

In this section, we shall present hardness proofs for two intermediate problems. We will then subsequently use these problems as reduction targets in our proofs of Theorems 2 and 3. We first explicitly generalise Definitions 5 and 6 to multigraphs in the natural way. We avoided doing so in the introduction because the graph separator problems that we have previously defined are trivially equivalent to their multigraph variants — we will only use these definitions for intermediate problems.

Definition 10. *Let $G = (V, E)$ be a multigraph, and let $s, t \in V(G)$. A multiset $F \subseteq E$ is an (s,t)-*edge separator *of G if s and t lie in different components of $G - F$. We say F is a* minimal (s,t)-edge separator *if no proper submultiset of F is an (s,t)-edge separator, and write* $\mathrm{MES}(G, s, t)$ *for the number of minimal (s,t)-edge separators of G.*

Definition 11. *Let $G = (V, E)$ be a multigraph, and let $F \subseteq E$. We say F is a* minimal edge separator *if it is a minimal (s,t)-edge separator for some $s, t \in V$, and write* $\mathrm{MES}(G)$ *for the number of minimal edge separators of G.*

We now define our two intermediate problems.

Problem 14. #LargeMinimalEdgeSeps.
Input: *A multigraph G and the maximum cardinality x of any minimal edge separator in G.*
Output: *The number of minimal edge separators of G with maximum cardinality, which we denote by* $\mathrm{LMES}(G)$.

Problem 15. #(s,t)-LargeMinimalEdgeSeps.
Input: *A multigraph G, two distinct vertices $s,t \in V$, and the maximum cardinality y of any minimal (s,t)-edge separator in G.*
Output: *The number of minimal (s,t)-edge separators of G with maximum cardinality, which we denote by* $\mathrm{LMES}(G,s,t)$.

Note that the input restrictions in the definitions of #LargeMinimalEdgeSeps and #(s,t)-LargeMinimalEdgeSeps are motivated purely by their uses as intermediate problems in reductions. When we use them, we will be able to prove that their respective promises are satisfied. As the next proposition shows, both #LargeMinimalEdgeSeps and #(s,t)-LargeMinimalEdgeSeps can be expressed in terms of vertex cuts. It is a widely known result and was first proved by Whitney [33] — we give a proof here for completeness.

Proposition 12. *Let $G = (V,E)$ be a connected multigraph. Then a multiset $F \subseteq E$ is a minimal edge separator of G if and only if $G - F$ has exactly two non-empty components, and F is the multiset of edges between them.*

Proof. For any non-empty set $S \subset V$ such that $G[S]$ and $G[V \setminus S]$ are connected, taking an arbitrary $s \in S$ and $t \in V \setminus S$, it is immediate that the multiset of edges between S and $V \setminus S$ is a minimal (s,t)-edge separator and hence a minimal edge separator.

Conversely, let $F \subseteq E(G)$ be a minimal (s,t)-edge separator for some $s,t \in V$. Suppose $G-F$ has (at least) three components C_1, C_2 and C_3. Without loss of generality, suppose $s \in C_1$ and $t \in C_2$. Then since G is connected, F must contain an edge e from $C_1 \cup C_2$ to C_3. But then $F \setminus \{e\}$ is still an (s,t)-edge separator, contradicting minimality. Hence $G - F$ has only two components, as required. □

Thus we may view counting maximum minimal edge separators as counting maximum vertex cuts subject to the requirement that each part of the vertex cut is connected. We shall therefore prove hardness for #LargeMinimalEdgeSeps and #(s,t)-LargeMinimalEdgeSeps by adapting a folklore proof that MAX-CUT is NP-complete (see e.g. Exercise 7.25 of Sipser [29]). The original proof works by reduction from 3-NAE-SAT – we shall instead reduce from the following variant of the problem.

Definition 13. *We define* NAE *to be a logical clause as follows. Let x_1, x_2 and x_3 be literals, and let $\sigma : \{x_1, x_2, x_3\} \to \{0,1\}$ be a truth assignment. Then under σ,*

$$\mathrm{NAE}(x_1,x_2,x_3) = \begin{cases} 0 & \text{if } \sigma(x_1) = \sigma(x_2) = \sigma(x_3) \\ 1 & \text{otherwise.} \end{cases}$$

Definition 14. *We define a* monotone 3-NAE *formula ϕ to be any logical formula of the form $\bigwedge_{i\in[k]} \mathcal{C}_i$, where $k \in \mathbb{N}$ and $\mathcal{C}_1,\dots,\mathcal{C}_k$ are* NAE *clauses containing three distinct and un-negated literals, e.g. three distinct variables.*

Problem 16. #MonotonePromise-3-NAE-SAT.
Input: *A satisfiable monotone* 3-NAE *formula* ϕ.
Output: *The number of satisfying assignments of* ϕ.

We first prove hardness for #MonotonePromise-3-NAE-SAT by reduction from #IS.

Lemma 15. #MonotonePromise-3-NAE-SAT *is* #SAT-*hard to approximate and is* #P-*complete.*

Proof. For every instance ϕ of #MonotonePromise-3-NAE-SAT, let $\mathrm{SAT}(\phi)$ be the number of satisfying assignments of ϕ. Since #MonotonePromise-3-NAE-SAT is in #P, we have #MonotonePromise-3-NAE-SAT $\leq_{\mathrm{AP}}$ #SAT by [10]. Let $G = (V, E)$ be an instance of #IS, which is hard by Theorem 7. We shall construct an instance ϕ of #MonotonePromise-3-NAE-SAT with the property that $\mathrm{SAT}(\phi) = 2 \cdot \mathrm{IS}(G)$, from which the result follows immediately.

We identify V with a set of logical variables. Let x be a new variable distinct from the variables in V. Then we define

$$\phi = \bigwedge_{\{i,j\} \in E} \mathrm{NAE}(i, j, x).$$

Note that ϕ is satisfiable by setting x to 1 and all other variables to 0, so ϕ is an instance of #MonotonePromise-3-NAE-SAT.

Suppose $\sigma : V \cup \{x\} \to \{0, 1\}$ is a satisfying assignment of ϕ. Then we may define an independent set S as follows.

$$S = \{v \in V \mid \sigma(v) = \sigma(x)\}.$$

Since σ is a satisfying assignment, we cannot have $\sigma(i) = \sigma(j) = \sigma(x)$ for any $\{i, j\} \in E$, and so S is an independent set.

Conversely, suppose S is an independent set of G and let 1_S be the indicator function of S. Then S corresponds to two satisfying assignments $\sigma_0, \sigma_1 : V \cup \{x\} \to \{0, 1\}$ of ϕ. Indeed, let $\sigma_1(x) = 1$, and let $\sigma_1(v) = 1_S(v)$ for all $v \in V$. Then σ_1 satisfies every clause $\mathrm{NAE}(i, j, x)$ of ϕ, since $\sigma_1(x) = 1$ and at most one of i and j lies in S. We then define $\sigma_0 = 1 - \sigma_1$, which is a satisfying assigmnent since σ_1 is a satisfying assignment.

Thus each satisfying assignment of ϕ corresponds to a unique independent set of G, and each independent set of G corresponds to exactly two satisfying assignments of ϕ. The result therefore follows. □

We now reduce #MonotonePromise-3-NAE-SAT to #LargeMinimalEdgeSeps and #(s, t)-LargeMinimalEdgeSeps.

Lemma 16. #LargeMinimalEdgeSeps *and* #(s, t)-LargeMinimalEdgeSeps *are* #SAT-*hard to approximate and are* #P-*complete.*

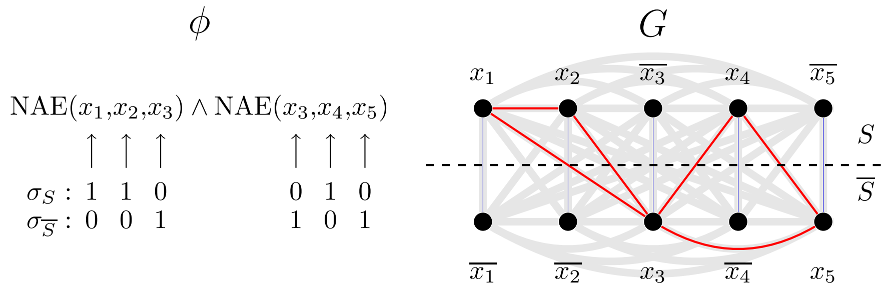

Figure 2: An example of the reduction from an instance ϕ of #MonotonePromise-3-NAE-SAT to an instance (G, k) of #LargeMinimalEdgeSeps used in the proof of Lemma 16. The thin blue edges of G are elements of F_1, the thick red edges are elements of F_2, and the very thick grey edges are elements of F_3. In this example we have $k = 5 + 4 + 5 \cdot 5 = 34$, and a minimal edge separator is maximum if and only if it contains all edges of F_1 and 4 edges of F_2.

Proof. Since #LargeMinimalEdgeSeps and #(s,t)-LargeMinimalEdgeSeps are in #P, it follows that #LargeMinimalEdgeSeps $\leq_{\mathrm{AP}}$ #SAT and #(s,t)-LargeMinimalEdgeSeps $\leq_{\mathrm{AP}}$ #SAT by [10]. We will first prove the result for #LargeMinimalEdgeSeps. Let ϕ be an instance of #MonotonePromise-3-NAE-SAT, which is hard by Lemma 15. Let $x_1, \ldots, x_n$ be the variables of ϕ, and let $\mathcal{C}_1, \ldots, \mathcal{C}_m$ be the clauses of ϕ. We shall construct an instance (G, k) of #LargeMinimalEdgeSeps with the property that $\mathrm{SAT}(\phi) = 2 \cdot \mathrm{LMES}(G)$, from which the result follows immediately. See Figure 2 for an example.

We define $G = (V, E)$ as follows. Let

$$V = \{x_i \mid i \in [n]\} \cup \{\overline{x_i} \mid i \in [n]\}.$$

Let $C_i \subseteq V$ be the set of variables appearing in clause $\mathcal{C}_i$. We now define sets of edges

$$\begin{aligned} F_1 &= \{\{x_i, \overline{x_i}\} \mid i \in [n]\}, \\ F_2 &= \biguplus_{i \in [m]} C_i^{(2)}, \\ F_3 &= V^{(2)}. \end{aligned}$$

We then define $E = F_1 \uplus F_2 \uplus F_3$. Finally, let $k = n + 2m + n^2$.

Suppose that F is a minimal edge separator of G. By Proposition 12, $G - F$ has exactly two components S and $V \setminus S$. We claim that $|F| \leq k$, with equality if and only if the following properties hold.

(i) For all $i \in [n]$, $|\{x_i, \overline{x_i}\} \cap S| = 1$.
(ii) For all $i \in [m]$, $|F \cap C_i^{(2)}| = 2$.

First, note that $|F \cap F_1| \leq n$ with equality if and only if (i) holds. Second, note that for all $i \in [m]$, we have $|F \cap C_i^{(2)}| \leq 2$ with equality for all i if and only if (ii) holds. Finally, note that

$$|F \cap F_3| = |S|(2n - |S|) = n^2 - (n - |S|)^2 \leq n^2$$

with equality if and only if $|S| = n$ (which is implied by (i)). Hence

$$|F| = |F \cap F_1| + \sum_{i \in [m]} |F \cap C_i^{(2)}| + |F \cap F_3| \leq k,$$

with equality if and only if (i) and (ii) hold. We will soon see that satisfying assignments of ϕ correspond to minimal edge separators satisfying (i) and (ii). Since ϕ is satisfiable, this will imply in particular that (G, k) is an instance of #LargeMinimalEdgeSeps.

We now define a two-to-one correspondence between satisfying assignments of ϕ and minimal edge separators of G of cardinality k. Given a satisfying assignment $\sigma : \{x_1, \dots, x_n\} \to \{0, 1\}$, let $S = \{x_i \mid \sigma(x_i) = 1\} \cup \{\overline{x_i} \mid \sigma(x_i) = 0\}$ and let $f(\sigma)$ be the multiset of edges from S to $V \setminus S$. Note that since G contains a spanning clique it is immediate that $f(\sigma)$ is a minimal $(x_1, \overline{x_1})$-edge separator, and hence a minimal edge separator. Moreover, since σ is a satisfying assignment, $f(\sigma)$ satisfies (i) and (ii) and therefore has cardinality k. It is immediate that f is a two-to-one map, with $f(\sigma) = f(1 - \sigma)$. It remains only to prove that f is surjective.

Let F be a minimal edge separator of G of cardinality k, let S be a component of $G - F$, and let $\sigma_S : \{x_1, \dots, x_n\} \to \{0, 1\}$ be given by

$$\sigma_S(x_i) = \begin{cases} 1 & \text{if } x_i \in S, \\ 0 & \text{if } \overline{x_i} \in S. \end{cases}$$

By property (i), σ_S is well-defined. Let $\mathcal{C}_i$ be a clause of ϕ. Then $C_i \cap S$ is the set of literals in $\mathcal{C}_i$ which are true under σ_S, and so σ_S satisfies $\mathcal{C}_i$ by property (ii). Hence σ_S is a satisfying assignment of ϕ, and so $\mathrm{SAT}(\phi) = 2 \cdot \mathrm{LMES}(G)$ as required.

Note that any maximum minimal edge separator in G is a maximum minimal $(x_1, \overline{x_1})$-edge separator and vice versa, and so we also have $\mathrm{SAT}(\phi) = 2 \cdot \mathrm{LMES}(G, x_1, \overline{x_1})$. The result therefore follows for $\#(s,t)$-LargeMinimalEdgeSeps as well. $\square$

4.2. Hardness of minimal separator problems

The remaining reductions necessary to prove Theorems 2 and 3 are all quite similar. For convenience, we combine them into the following two lemmas. The first lemma will be used to prove Theorem 3.

Lemma 17. *Let $G = (V, E)$ be a connected multigraph, writing $n = |V|$ and $m = |E|$. Suppose (G, x) is an instance of* #LargeMinimalEdgeSeps, *and (G, s, t, y) is an instance of* $\#(s,t)$-LargeMinimalEdgeSeps. *Let $k = \lceil m + \log_2(m) + 10 \rceil$. Then there exists a graph G' such that the following properties hold.*

(i) *G' is bipartite, $V \subseteq V(G')$, and $|V(G')| \leq |E|k + |V|$.*
(ii) $\mathrm{LMES}(G) \leq \mathrm{MES}(G')/2^{kx} \leq \mathrm{LMES}(G) + \frac{1}{4}$.
(iii) $\mathrm{LMES}(G,s,t) \leq \mathrm{MES}(G',s,t)/2^{ky} \leq \mathrm{LMES}(G,s,t) + \frac{1}{4}$.

Proof. Informally, we form G' by first k-thickening and then 2-stretching each edge of G. Formally, we define G' as follows. For each $e \in E$ let X_e be a set of k vertices, disjoint from V, where $X_e \cap X_f = \emptyset$ whenever $e \neq f$. Let $X = \bigcup_{e \in E} X_e$. Then we define

$$\begin{aligned} V(G') &= V \cup X, \\ E(G') &= \bigcup_{e=\{u,v\} \in E} \{\{u,w\},\{w,v\} \mid w \in X_e\}. \end{aligned}$$

Thus G' satisfies property (i). For each $e = \{u,v\} \in E$, let $P_1^e, \dots, P_k^e$ be the internally vertex-disjoint paths in G' of the form uwv with $w \in X_e$.

We say a minimal edge separator F' of G' is *good* if it is not of the form $E(P_i^e)$ for some $e \in E$, $i \in [k]$. Note that every good minimal edge separator F' of G' satisfies the following properties.

(a) $|F' \cap E(P_i^e)| \leq 1$ for all $e \in E$, $i \in [k]$.
(b) If $|F' \cap E(P_i^e)| = 1$ for some $e \in E$, $i \in [k]$, then $|F' \cap E(P_j^e)| = 1$ for all $j \in [k]$.

For a good minimal edge separator F' of G', write

$$\pi(F') = \{e \in E \mid F' \cap E(P_i^e) \neq \emptyset \text{ for some } i \in [k]\}.$$

We say that a minimal edge separator F of G corresponds to a good minimal edge separator F' of G' when $F = \pi(F')$. By properties (a) and (b), any minimal edge separator F of G corresponds to exactly $2^{k|F|}$ good minimal edge separators of G'. Conversely, any good minimal edge separator of G' corresponds to a single minimal edge separator of G. Finally, there are exactly mk non-good minimal edge separators of G'. Hence, writing $\mathrm{MES}_i(G)$ for the number of minimal edge separators of G with cardinality i, we have

$$\mathrm{MES}(G') = \sum_{i=1}^{x} \mathrm{MES}_i(G) \cdot 2^{ki} + mk.$$

It follows immediately that $\mathrm{MES}(G')/2^{kx} \geq \mathrm{LMES}(G)$. Moreover, we have

$$\begin{aligned} \mathrm{MES}(G')/2^{kx} &= \mathrm{LMES}(G) + \sum_{i=1}^{x-1} \mathrm{MES}_i(G) \cdot 2^{k(i-x)} + mk \cdot 2^{-kx} \\ &\leq \mathrm{LMES}(G) + m \cdot 2^m \cdot 2^{-k} + k^2 \cdot 2^{-k} \\ &\leq \mathrm{LMES}(G) + \frac{1}{8} + \frac{1}{8} = \mathrm{LMES}(G) + \frac{1}{4}. \end{aligned}$$

(In the penultimate inequality, we use the fact that G is connected and so $x \geq 1$. In the final inequality, we use the fact that $k \geq 10$ and hence $k^2 \cdot 2^{-k} \leq 1/8$.)

Hence G' satisfies property (ii). Moreover, minimal (s,t)-edge separators of G correspond only to good minimal (s,t)-edge separators of G' and vice versa, and so G' satisfies property (iii) by the same argument. $\square$

We can now prove Theorem 3.

Theorem 3. *The problems* #BiMinimalEdgeSeps *and* #(s,t)-BiMinimalEdgeSeps *are* #P*-complete and are equivalent to* #SAT *under AP-reduction.*

Proof. Both problems are in #P, and hence AP-reducible to #SAT by [10]. As in the proof of Theorem 1, Lemma 17 implies that

$$\#\text{LargeMinimalEdgeSeps} \leq_{\text{AP}} \#\text{BiMinimalEdgeSeps},$$
$$\#(s,t)\text{-LargeMinimalEdgeSeps} \leq_{\text{AP}} \#(s,t)\text{-BiMinimalEdgeSeps}.$$

Moreover, since $\text{LMES}(G)$ and $\text{LMES}(G,s,t)$ are integers for all G, s and t, Lemma 17 also yields exact Turing reductions. The result therefore follows by Lemma 16. $\square$

The second lemma will be used to prove Theorem 2.

Lemma 18. *Let $G = (V,E)$ be a connected multigraph, writing $n = |V|$ and $m = |E|$. Suppose (G,x) is an instance of* #LargeMinimalEdgeSeps*, and (G,s,t,y) is an instance of* #(s,t)-LargeMinimalEdgeSeps*. Let $k = \lceil m + n + \log_3(n^2) + 16\rceil$. Then there exists a graph G' such that the following properties hold.*

(i) *G' is bipartite, $V \subseteq V(G')$, and $|V(G')| \leq 3|E|k + |V|$.*
(ii) $\text{LMES}(G) \leq \text{MS}(G')/3^{kx} \leq \text{LMES}(G) + \frac{1}{4}$.
(iii) $\text{LMES}(G,s,t) \leq \text{MS}(G',s,t)/3^{ky} \leq \text{LMES}(G,s,t) + \frac{1}{4}$.
(iv) $\text{LMES}(G) \leq \text{IMS}(G')/3^{kx} \leq \text{LMES}(G) + \frac{1}{4}$.

Proof. Informally, we form G' by first k-thickening and then 4-stretching each edge of G. Formally, for each $e \in E$, let X^e, Y^e and Z^e be sets of k vertices. We require all of these sets to be disjoint from each other and from V. For each $e \in E$, write $X^e = \{x_1^e, \dots, x_k^e\}$, $Y^e = \{y_1^e, \dots, y_k^e\}$ and $Z^e = \{z_1^e, \dots, z_k^e\}$. Write $W^e = X^e \cup Y^e \cup Z^e$, and $W = \bigcup_{e \in E} W^e$. Arbitrarily labelling e's endpoints as u and v, for each $i \in [k]$ let P_i^e be the path $ux_i^e y_i^e z_i^e v$. Thus the paths $P_1^e, \dots, P_k^e$ are k internally vertex-disjoint paths of length 4 between e's endpoints with $V(P_i^e) = \{u, x_i^e, y_i^e, z_i^e, v\}$. Then we define

$$V(G') = V \cup W,$$
$$E(G') = \bigcup_{\substack{e \in E \\ i \in [k]}} E(P_i^e).$$

It is immediate that G' satisfies property (i).

We will be able to associate minimal separators of G' with minimal edge separators of G in much the same way as in the proof of Lemma 17, but the correspondence will be messier since a minimal separator of G' may contain vertices of V. Indeed, there may be exponentially many such separators in k.

We define our correspondence as follows. If X is a minimal separator of G', we write

$$\pi(X) = \{e \in E \mid X \cap W^e \neq \emptyset\}.$$

We say a minimal separator X of G' is *z-good*, where $z \in \mathbb{N}$, if it satisfies the following conditions.

(a) We have $|X \cap V(P_i^e)| \leq 1$ for all $e \in E$, $i \in [k]$.
(b) Whenever $|X \cap V(P_i^e)| = 1$ for some $e \in E$ and $i \in [k]$, we have $|X \cap V(P_j^e)| = 1$ for all $j \in [k]$.
(c) We have $X \cap V = \emptyset$.
(d) We have $|\pi(X)| = z$.

We say that X is *good* if it is z-good for some $z \in \mathbb{N}$.

Claim 1. *All but at most $3^{kx}/4$ minimal separators of G' are x-good, and all but at most $3^{ky}/4$ minimal (s,t)-separators of G' are y-good.*

We shall defer the proof of Claim 1 for the moment. We say that each good minimal separator X of G' corresponds to the multiset $\pi(X) \subseteq E$. Note that any minimal edge separator F of G corresponds to exactly $3^{k|F|}$ good minimal separators of G', all of which are $|F|$-good by the definition of z-goodness. Conversely each z-good minimal separator of G' corresponds to a single multiset $F \subseteq E$, which is a minimal edge separator of G with cardinality z. Hence by Claim 1,

$$\mathrm{LMES}(G) \cdot 3^{kx} \leq \mathrm{MS}(G') \leq \mathrm{LMES}(G) \cdot 3^{kx} + \frac{3^{kx}}{4}.$$

Hence (ii) is satisfied. Moreover, good minimal (s,t)-separators of G' correspond to minimal (s,t)-edge separators of G and vice versa, so (iii) is likewise satisfied by Claim 1.

Finally, we claim that the following holds.

$$\text{Every good minimal separator } X \text{ of } G' \text{ separates } G' - X \text{ into exactly two components.} \tag{2}$$

Indeed, $\pi(X)$ is a minimal edge separator of G, and so by Proposition 12 $G - \pi(X)$ has exactly two components. Since X is good, it follows that $G' - X$ has exactly two components also. Hence (2) holds. In particular, this implies that every good minimal separator of G' is inclusion-minimal, and so (iv) is satisfied.

It remains only to prove Claim 1. We shall first prove that most minimal separators of G' are minimal (b,c)-separators for some $b, c \in V$ (see Subclaim 1). We shall then prove that most such minimal separators X of G' maximise $|\pi(X)|$ (see Subclaim 2). Finally, we shall prove that if X does maximise $|\pi(X)|$ then X is good (see Subclaim 3). The first part of Claim 1 will therefore follow easily. Moreover, Subclaims 2 and 3 will imply the second part of Claim 1 in a similar fashion.

Subclaim 1. *There are at most 2^5mk minimal separators in G' which are not minimal (b,c)-separators for some $b,c \in V$.*

Proof of Subclaim 1: Let X be a minimal (b,c)-separator in G' for some $b,c \in V(G')$. We say X is *trivial* if $X \subseteq V(P_i^e)$ for some $e \in E$, $i \in [k]$. We claim that if X is not a (b',c')-separator for some $b',c' \in V$ then X is trivial, from which the result follows.

Suppose without loss of generality that b is an internal vertex of P_i^e for some $e \in E$, $i \in [k]$. Suppose that the component of $G' - X$ containing b is a subset of W^e. Then $X \cap V(P_i^e)$ is already a (b,c)-separator, and so by minimality we have $X \subseteq V(P_i^e)$. Hence X is trivial. We may therefore assume that the component of $G' - X$ containing b also contains some endpoint $b' \in V$ of e.

If $c \in V$ then X is a minimal (b',c)-separator and we are done. If c is an internal vertex of $V(P_j^f)$ for some $f \in E$, $j \in [k]$, then by the same argument either X is trivial or there exists $c' \in V$ such that c' and c lie in the same component of $G' - X$. Thus X is either trivial or a minimal (b',c')-separator, as required. We have therefore proved Subclaim 1.

Subclaim 2. *Let $a \in \mathbb{N}$, and let $b,c \in V$ be distinct. There are at most $2^{m+n}3^{k(a-1)}$ minimal (b,c)-separators X of G' with $|\pi(X)| < a$.*

Proof of Subclaim 2: We may choose any minimal (b,c)-separator X of G' by choosing first $X \cap V$, then $\pi(X)$, then $X \cap W^e$ for each $e \in \pi(X)$. There are at most 2^n ways of choosing $X \cap V$ and at most 2^m ways of choosing $\pi(X)$. For each $e \in \pi(X)$, since $b,c \in V$, X must contain exactly one vertex internal to each P_i^e and so there are exactly 3^k ways of choosing $X \cap W^e$. Since $|\pi(X)| \leq a-1$, Subclaim 2 follows.

Subclaim 3. *Let $b,c \in V$ be distinct, and let z be the maximum cardinality of any minimal (b,c)-edge separator of G. If X is a minimal (b,c)-separator of G' with $|\pi(X)| \geq z$, then X is z-good.*

Proof of Subclaim 3: Note that since $b,c \in V$, if $X \cap W^e \neq \emptyset$ for some $e \in E$ then $|X \cap \{x_i^e, y_i^e, z_i^e\}| = 1$ for all $i \in [k]$. In particular, if X satisfies (c) then X satisfies (a) and (b). To prove that X satisfies (c) and (d), we shall exhibit a minimal (b,c)-edge separator F of G with cardinality at least $|\pi(X)| + |X \cap V|$. Thus (c) and (d) will follow from the definition of z and the fact that $|\pi(X)| \geq z$. See Figure 3 for an example.

We say a pair (Y,D) with $Y \subseteq V$ and $D \subseteq E$ is a *hybrid minimal (b,c)-separator* of G if it satisfies the following properties.

(P1) b and c lie in separate components of $(G-D)-Y$.
(P2) For all $Y' \subset Y$, b and c lie in the same component of $(G-D)-Y'$.
(P3) For all $D' \subset D$, b and c lie in the same component of $(G-D')-Y$.

Thus $(X \cap V, \pi(X))$ is a hybrid minimal (b,c)-separator of G, since X is a minimal (b,c)-separator of G'.

Let C be the component of $(G-\pi(X)) - (X \cap V)$ containing b. For each $v \in X \cap V$, let $F_v \subseteq E$ be the multiset of edges between v and C in G. Let

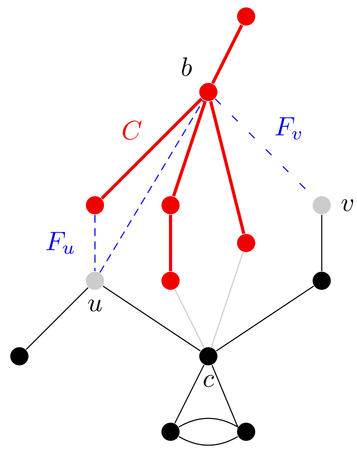

Figure 3: An example of the minimal (b,c)-edge separator F of G formed in the proof of Subclaim 3. The grey vertices and edges are elements of the hybrid minimal (b,c)-separator of G corresponding to X. Thus F consists of the grey edges of G together with the edges in F_u and F_v.

$F = \pi(X) \cup \bigcup_{v \in X \cap V} F_v$. We claim that F is the required minimal (b,c)-edge separator of G.

Note that $F_u \cap F_v = \emptyset$ for all distinct $u, v \in X \cap V$. For all $v \in X \cap V$, we must have $F_v \neq \emptyset$ or (P2) would be violated on taking $Y' = (X \cap V) \setminus \{v\}$. Moreover, we must have $F_v \cap \pi(X) = \emptyset$ or (P3) would be violated on taking $D' = \pi(X) \setminus F_v$. Hence $|F| \geq |\pi(X)| + |X \cap V|$ as required.

It is immediate from (P1) that F is a (b,c)-edge separator of G. Finally, note that F is minimal — (P2) implies that no edge in any F_v can be removed from F, and (P3) implies that no edge in $\pi(X)$ can be removed from F. Thus F is a minimal (b,c)-edge separator of cardinality at least $|\pi(X)|+|X \cap V|$, and so Subclaim 3 follows.

We now prove the first part of Claim 1. By Subclaim 1, all but at most 2^5mk minimal separators of G' are minimal (b,c)-separators for some $b, c \in V$. Moreover, by Subclaim 2, there are at most $n^2 \cdot 2^{m+n}3^{k(x-1)}$ such separators X with $|\pi(X)| < x$. Finally, by Subclaim 3 and the definition of x, if $|\pi(X)| \geq x$ then X is x-good. It follows that all but at most $2^5mk+n^2 2^{m+n}3^{k(x-1)}$ minimal separators of G' are x-good. We have

$$2^5mk \leq 2^5k^2 \leq 2^{k+5} \leq 3^{\frac{2}{3}(k+5)} \leq 3^{k-2} \leq \frac{3^{kx}}{8}$$

and

$$n^2 2^{m+n} 3^{k(x-1)} \leq 3^{k-16} 3^{k(x-1)} \leq \frac{3^{kx}}{8},$$

so all but at most $3^{kx}/4$ minimal separators of G' are x-good as required.

The second part of Claim 1 follows more easily. By Subclaim 2, all but at most $2^{m+n}3^{k(y-1)}$ minimal (s,t)-separators X of G' satisfy $|\pi(X)| \geq y$. It therefore follows from Subclaim 3 that all but at most

$$2^{m+n}3^{k(y-1)} \leq \frac{3^{ky}}{4}$$

minimal (s,t)-separators of G' are y-good as required. Thus Claim 1 follows, as does the result. $\square$

We can now prove Theorem 2.

Theorem 2. *The problems* #(s,t)-BiMinimalSeps*,* #BiMinimalSeps*,* #(s,t)-BiConnMinimalSeps*,* #BiConnMinimalSeps *and* #BiInclusionMinimalSeps *are* #P*-complete and are equivalent to* #SAT *under AP-reduction.*

Proof. All five problems are in #P, and hence AP-reducible to #SAT by [10]. As in the proof of Theorem 1, Lemma 18 implies that

$$\begin{aligned}
\#\text{LargeMinimalEdgeSeps} &\leq_{\text{AP}} \#\text{BiMinimalSeps},\\
\#\text{LargeMinimalEdgeSeps} &\leq_{\text{AP}} \#\text{BiInclusionMinimalSeps},\\
\#(s,t)\text{-LargeMinimalEdgeSeps} &\leq_{\text{AP}} \#(s,t)\text{-BiMinimalEdgeSeps}.
\end{aligned}$$

Moreover, since $\text{LMES}(G)$ and $\text{LMES}(G,s,t)$ are integers for all G, s and t, Lemma 18 also yields exact Turing reductions. Finally, note that in the proof of Lemma 18, all good minimal separators of G' separate G' into two components (see (2)). Analogues of Lemma 18(ii)–(iv) for #(s,t)-BiConnMinimalSeps and #BiConnMinimalSeps therefore follow instantly. The result now follows by Lemma 16. $\square$

5. Problems related to #MaximalBIS

5.1. Hardness of #BiDomSets

Recall that #MaximalBIS can be viewed as counting the number of independent dominating sets in a bipartite graph — a combination of #BIS and #BiDomSets. We shall now prove that #BiDomSets is #SAT-hard. We shall reduce from the following problem, which is well-known to be #SAT-hard in the guise of #IS (see Theorem 7).

Definition 19. *Let $G = (V,E)$ be a graph. A set $S \subseteq V$ is a* vertex cover *of G if $e \cap S \neq \emptyset$ for all $e \in E$. We write* $\text{VC}(G)$ *for the number of vertex covers of G.*

Problem 17. #VertexCovers*.*
Input: *A graph G.*
Output: *The number of vertex covers of G, which we denote by* $\text{VC}(G)$*.*

We can now prove Theorem 4.

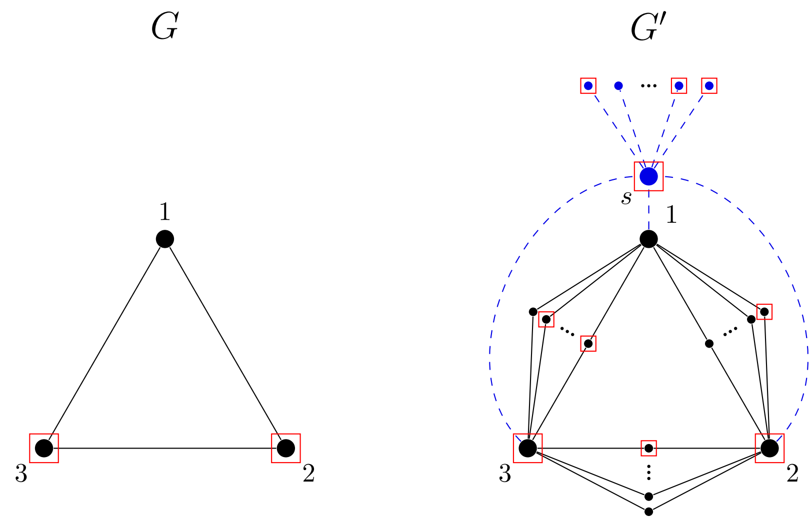

Figure 4: An example of the reduction from an instance G of #VertexCovers to an instance G' of #BiDomSets used in the proof of Theorem 4. The boxes around vertices indicate a vertex cover in G and a corresponding dominating set in G'. Note in particular how the presence of s ensures that vertices 1, 2 and 3 are dominated in G'.

Theorem 4. #BiDomSets $\equiv_{\mathrm{AP}}$ #SAT.

Proof. For every instance G' of #BiDomSets, let $\mathrm{DS}(G')$ be the number of dominating sets in G'. Since #BiDomSets is in #P, #BiDomSets $\leq_{\mathrm{AP}}$ #SAT follows from [10]. We will show #VertexCovers $\leq_{\mathrm{AP}}$ #BiDomSets. Let $G = (V, E)$ be an instance of #VertexCovers. Without loss of generality, let $V = [n]$ for some $n \in \mathbb{N}$, let $m = |E|$, and let $t = \lceil n + \log_2(m+1) + 3 \rceil$. We shall construct an instance G' of #BiDomSets with the property that $\mathrm{VC}(G) \leq \mathrm{DS}(G')/2^{(m+1)t} \leq \mathrm{VC}(G) + \frac{1}{4}$, which will be sufficient for the reduction as in the proof of Theorem 1. See Figure 4 for an example.

Informally, we obtain a bipartite graph G' (an instance of #BiDomSets) from G by first thickening and then 2-stretching each edge, then adding a gadget to G's vertices which will allow us to ignore their domination constraints. Formally, we define G' as follows. For each $e \in E$ let X_e be a set of t vertices, disjoint from $[n]$, where $X_e \cap X_f = \emptyset$ whenever $e \neq f$. Let $W = \bigcup_{e \in E} X_e$. Let Y be a set of t vertices disjoint from $[n] \cup W$, and let s be a vertex not contained in $[n] \cup W \cup Y$. Then we define

$$V(G') = Y \cup \{s\} \cup [n] \cup W,$$
$$E(G') = \{\{y, s\} \mid y \in Y\} \cup \{\{s, i\} \mid i \in [n]\} \cup \bigcup_{e=\{i,j\} \in E} \{\{i, x\}, \{x, j\} \mid x \in X_e\}.$$

We say a dominating set $S \subseteq V(G')$ is *good* if the following conditions hold.

(i) $s \in S$.

(ii) For all $e \in E$, we have $e \cap S \neq \emptyset$.

We will show that good dominating sets in G' correspond to vertex covers in G, and that almost all dominating sets in G' are good.

First note that there are exactly $2^{(m+1)t}$ ways of extending any vertex cover X of G into a good dominating set of G'. Indeed, a set S satisfying $X \cap [n] = S$ is a good dominating set of G' if and only if $s \in S$. Hence there are $2^{(m+1)t}\mathrm{VC}(G)$ good dominating sets of G', and in particular $\mathrm{DS}(G')/2^{(m+1)t} \geq \mathrm{VC}(G)$.

Moreover, suppose that S is a dominating set of G' which is not good. Then either $s \notin S$ or $e \cap S = \emptyset$ for some $e \in E$. If $s \notin S$, then $Y \subseteq S$. If $e \cap S = \emptyset$ for some $e \in E$, then $X_e \subseteq S$. Note that $n + \log_2(m+1) + 2 \leq t$, so $2^{mt+n+\log_2(m+1)} \leq 2^{(m+1)t-2}$. Hence there are at most

$$(m+1)2^{|V(G')|-t} = (m+1)2^{mt+n+1} \leq \frac{2^{(m+1)t}}{4}$$

dominating sets of G' which are not good. In particular, we have

$$\frac{\mathrm{DS}(G')}{2^{(m+1)t}} \leq \mathrm{VC}(G) + \frac{1}{4}.$$

The result therefore follows. □

5.2. Hardness of #SetUnions

We shall now prove that #SetUnions is #SAT-hard by a reduction from #MaximalBIS.

Theorem 5. #SetUnions $\equiv_{\mathrm{AP}}$ #SAT.

Proof. Since #SetUnions is in #P, #SetUnions $\leq_{\mathrm{AP}}$ #SAT follows from [10]. We will show #MaximalBIS $\leq_{\mathrm{AP}}$ #SetUnions. Let $G = (V, E)$ be an instance of #MaximalBIS with vertex classes A and B. Note that #MaximalBIS is hard by Theorem 1. Without loss of generality, let $A = [n]$ for some $n \in \mathbb{N}$. We shall construct an instance $(n, \mathcal{F})$ of #SetUnions with the property that $\mathrm{MIS}(G) = |\mathcal{U}(\mathcal{F})|$, from which the result follows immediately. See Figure 5 for an example.

Let $\mathcal{F} = \{N(v) \mid v \in B\}$, so that

$$\mathcal{U}(\mathcal{F}) = \{N(S) \mid S \subseteq B\}. \tag{3}$$

Given $S \subseteq [n]$, write $\overline{S} = [n] \setminus S$. Similarly, given $S \subseteq B$, write $\overline{S} = B \setminus S$. Given $S \subseteq [n]$, write $\mathrm{MIS}_S(G)$ for the number of maximal independent sets $X \subseteq V$ with $X \cap [n] = S$. We shall prove that

$$\mathrm{MIS}_S(G) = \begin{cases} 1 & \text{if } \overline{S} \in \mathcal{U}(\mathcal{F}), \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

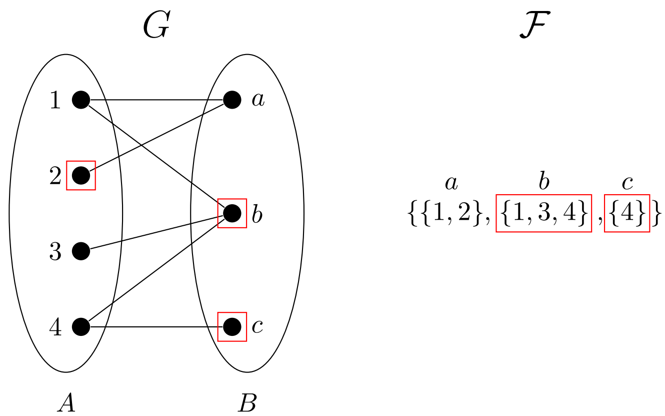

Figure 5: An example of the reduction from an instance G of #MaximalBIS to an instance $(n, \mathcal{F})$ of #SetUnions used in the proof of Theorem 5. The boxes around vertices indicate a maximal independent set in G and the corresponding union of sets in $\mathcal{F}$.

It will follow immediately that the map $X \mapsto \overline{X \cap [n]}$ is a bijection from the set of maximal independent sets of G to $\mathcal{U}(\mathcal{F})$.

Take $S \subseteq [n]$. Note that $\text{MIS}_S(G) \in \{0, 1\}$ — any maximal independent set $X \subseteq V$ is uniquely determined by its intersection with $[n]$. It therefore suffices to prove that $\text{MIS}_S(G) > 0$ if and only if $\overline{S} \in \mathcal{U}(\mathcal{F})$.

First suppose $\overline{S} \in \mathcal{U}(\mathcal{F})$. Let $T \subseteq B$ be a maximal set such that $\overline{S} = N(T)$. Then it is immediate that there are no edges between S and T. Moreover, $\overline{T} \subseteq N(S)$ by maximality of T. Hence $S \cup T$ is a maximal independent set in G, and $\text{MIS}_S(G) = 1$.

Now suppose $\overline{S} \notin \mathcal{U}(\mathcal{F})$, and suppose $X \subseteq V(G)$ is a maximal independent set of G with $X \cap [n] = S$. Then by independence we have $N(S) \cap X = \emptyset$, and by maximality we have $\overline{N(S)} \subseteq X$. Thus $X = S \cup \overline{N(S)}$. By maximality, it follows that $\overline{S} \subseteq N(\overline{N(S)})$ — otherwise an element of $\overline{S}$ could be added to X. But $N(\overline{N(S)}) \cap S = \emptyset$ since $\overline{N(S)}$ is precisely the set of vertices in B with no edges to S, so $N(\overline{N(S)}) \subseteq \overline{S}$ and hence $\overline{S} = N(\overline{N(S)})$. But this implies $\overline{S} \in \mathcal{U}(\mathcal{F})$ by equation (3), which is a contradiction. Hence $\text{MIS}_S(G) = 0$, and we have proved equation (4). It follows that $\text{MIS}(G) = |\mathcal{U}(\mathcal{F})|$, as required. $\square$

5.3. Hardness of #UnionReps

We shall now prove that #UnionReps is #SAT-hard by reducing from #VertexCovers.

Theorem 6. #UnionReps $\equiv_{\text{AP}}$ #SAT.

Proof. Since #UnionReps is in #P, #UnionReps $\leq_{\text{AP}}$ #SAT follows from [10]. We will show #VertexCovers $\leq_{\text{AP}}$ #UnionReps. Let $G = (V, E)$ be an instance of #VertexCovers, which is hard by Theorem 7. Without loss of generality let

$V = [n]$ for some $n \in \mathbb{N}$, and let $m = |E|$. We shall construct an instance $(m, \mathcal{F})$ of #UnionReps with the property that $|\mathcal{U}_{\mathcal{F}}^{-1}([m])| = \mathrm{VC}(G)$, from which the result follows immediately.

For each $i \in [n]$, let S_i be the set of edges incident to i in G. Let $\mathcal{F} = \{S_i \mid i \in [n]\}$. Thus on identifying E with $[m]$, $(m, \mathcal{F})$ becomes an instance of #UnionReps. Given a set $X \subseteq [n]$, let $X' = \{S_i : i \in X\}$. Then X is a vertex cover of G if and only if $\cup X' = E$. Thus there is a bijection between $\mathcal{U}_{\mathcal{F}}^{-1}(E)$ and vertex covers of G, and so $|\mathcal{U}_{\mathcal{F}}^{-1}(E)| = \mathrm{VC}(G)$ as required. $\square$

Index of problems

Acknowledgements

We thank Luca Manzoni and Yuri Pirola for useful discussions.

References

[1] Anne Berry, Jean Paul Bordat, and Olivier Cogis. Generating all the minimal separators of a graph. *International Journal of Foundations of Computer Science*, 11(3):397–403, 2000.

[2] Hans L. Bodlaender and Fedor V. Fomin. Tree decompositions with small cost. *Discrete Applied Mathematics*, 145(2):143 – 154, 2005.

[3] Hans L. Bodlaender, Fedor V. Fomin, Arie M.C.A. Koster, Dieter Kratsch, and Dimitrios M. Thilikos. On exact algorithms for treewidth. *ACM Trans. Algorithms*, 9(1):12:1–12:23, 2012.

[4] Hans L. Bodlaender and Arie M.C.A. Koster. Combinatorial optimization on graphs of bounded treewidth. *The Computer Journal*, 51(3):255–269, 2008.

[5] Vincent Bouchitté and Ioan Todinca. Treewidth and minimum fill-in: grouping the minimal separators. *SIAM Journal on Computing*, 31(1):212–232, 2001.

[6] Vincent Bouchitté and Ioan Todinca. Listing all potential maximal cliques of a graph. *Theoretical Computer Science*, 276(1-2):17–32, 2002.

[7] Henning Bruhn, Pierre Charbit, Oliver Schaudt, and Jan Arne Telle. The graph formulation of the union-closed sets conjecture. *European J. Combin.*, 43:210–219, 2015.

[8] Pierluigi Crescenzi. A short guide to approximation preserving reductions. In *Twelfth Annual IEEE Conference on Computational Complexity (Ulm, 1997)*, pages 262–273. IEEE Computer Soc., Los Alamitos, CA, 1997.

[9] Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.

[10] Martin Dyer, Leslie Ann Goldberg, Catherine Greenhill, and Mark Jerrum. The relative complexity of approximate counting problems. *Algorithmica*, 38(3):471–500, 2004. Approximation algorithms.

[11] Fedor V. Fomin, Dieter Kratsch, Ioan Todinca, and Yngve Villanger. Exact algorithms for treewidth and minimum fill-in. *SIAM Journal on Computing*, 38(3):1058–1079, 2008.

[12] Fedor V. Fomin and Yngve Villanger. Finding induced subgraphs via minimal triangulations. In *27th International Symposium on Theoretical Aspects of Computer Science, STACS 2010, March 4-6, 2010, Nancy, France*, pages 383–394, 2010.

[13] Fedor V. Fomin and Yngve Villanger. Treewidth computation and extremal combinatorics. *Combinatorica*, 32(3):289–308, 2012.

[14] Masanobu Furuse and Koichi Yamazaki. A revisit of the scheme for computing treewidth and minimum fill-in. *Theoretical Computer Science*, 531(0):66 – 76, 2014.

[15] Leslie Ann Goldberg. *Efficient Algorithms for Listing Combinatorial Structures*. Cambridge University Press, 1993. Cambridge Books Online.

[16] Leslie Ann Goldberg, Rob Gysel, and John Lapinskas. Approximately counting locally-optimal structures. In *Automata, Languages, and Programming: 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I*, pages 654–665, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[17] Leslie Ann Goldberg and Mark Jerrum. The complexity of ferromagnetic Ising with local fields. *Combin. Probab. Comput.*, 16(1):43–61, 2007.

[18] Leslie Ann Goldberg and Mark Jerrum. Approximating the partition function of the ferromagnetic Potts model. *J. ACM*, 59(5):Art. 25, 31, 2012.

[19] Leslie Ann Goldberg and Mark Jerrum. The complexity of approximately counting tree homomorphisms. *ACM Trans. Comput. Theory*, 2(2), 2014.

[20] Rob Gysel. Unique perfect phylogeny characterizations via uniquely representable chordal graphs. *CoRR*, abs/1305.1375, 2013.

[21] Rob Gysel. Minimal triangulation algorithms for perfect phylogeny problems. In *Proceedings of the 8th International Conference on Language and Automata Theory and Applications, Lecture Notes in Computer Science*, volume 8370, pages 421–432. Springer, 2014.

[22] David S. Johnson, Christos H. Papadimitriou, and Mihalis Yannakakis. How easy is local search? *J. Comput. Syst. Sci.*, 37(1):79–100, 1988.

[23] Toshinobu Kashiwabara, Sumio Masuda, Kazuo Nakajima, and Toshio Fujisawa. Generation of maximum independent sets of a bipartite graph and maximum cliques of a circular-arc graph. *J. Algorithms*, 13(1):161–174, 1992.

[24] Shuji Kijima, Yoshio Okamoto, and Takeaki Uno. Dominating set counting in graph classes. In *Computing and Combinatorics - 17th Annual International Conference, COCOON 2011, Dallas, TX, USA, August 14-16, 2011. Proceedings*, pages 13–24, 2011.

[25] Ton Kloks and Dieter Kratsch. Listing all minimal separators of a graph. *SIAM Journal on Computing*, 27:605–613, 1998.

[26] Daniel Lokshtanov. On the complexity of computing treelength. *Discrete Applied Mathematics*, 158(7):820 – 827, 2010.

[27] Alejandro A. Schäffer and Mihalis Yannakakis. Simple local search problems that are hard to solve. *SIAM J. Comput.*, 20(1):56–87, 1991.

[28] Hong Shen and Weifa Liang. Efficient enumeration of all minimal separators in a graph. *Theoretical Computer Science*, 180(1-2):169–180, 1997.

[29] Michael Sipser. *Introduction to the Theory of Computation.* Cengage Learning, 2nd edition, 2005.

[30] Ken Takata. Space-optimal, backtracking algorithms to list the minimal vertex separators of a graph. *Discrete Appl. Math.*, 158(15):1660–1667, 2010.

[31] Shuji Tsukiyama, Mikio Ide, Hiromu Ariyoshi, and Isao Shirakawa. A new algorithm for generating all the maximal independent sets. *SIAM J. Comput.*, 6(3):505–517, 1977.

[32] Salil P. Vadhan. The complexity of counting in sparse, regular, and planar graphs. *SIAM J. Comput.*, 31(2):398–427, 2001.

[33] Hassler Whitney. Planar graphs. *Fundamenta Mathematicae*, 21(1):73–84, 1933.

[34] David Zuckerman. On unapproximable versions of NP-complete problems. *SIAM J. Comput.*, 25(6):1293–1304, 1996.