

Fixed-Parameter and Approximation Algorithms: A New Look

Rajesh Chitnis *

MohammadTaghi Hajiaghayi*

Guy Kortsarz[†]

August 9, 2013

Abstract

A Fixed-Parameter Tractable (FPT) ρ -approximation algorithm for a minimization (resp. maximization) parameterized problem P is an FPT algorithm that, given an instance $(x, k) \in P$ computes a solution of cost at most $k \cdot \rho(k)$ (resp. $k/\rho(k)$) if a solution of cost at most (resp. at least) k exists; otherwise the output can be arbitrary. For well-known intractable problems such as the W[1]-hard Clique and W[2]-hard Set Cover problems, the natural question is whether we can get any FPT-approximation. It is widely believed that both Clique and Set-Cover admit no FPT ρ -approximation algorithm, for any increasing function ρ . However, to the best of our knowledge, there has been no progress towards proving this conjecture. Assuming standard conjectures such as the Exponential Time Hypothesis (ETH) [19] and the Projection Games Conjecture (PGC) [29], we make the first progress towards proving this conjecture by showing that

- Under the ETH and PGC, there exist constants $F_1, F_2 > 0$ such that the Set Cover problem does not admit a FPT approximation algorithm with ratio k^{F_1} in $2^{k^{F_2}} \cdot \text{poly}(N, M)$ time, where N is the size of the universe and M is the number of sets.
- Unless $\text{NP} \subseteq \text{SUBEXP}$, for every $1 > \delta > 0$ there exists a constant $F(\delta) > 0$ such that Clique has no FPT cost approximation with ratio $k^{1-\delta}$ in $2^{k^F} \cdot \text{poly}(n)$ time, where n is the number of vertices in the graph.

In the second part of the paper we consider various W[1]-hard problems such as Directed Steiner Tree, Directed Steiner Forest, Directed Steiner Network and Minimum Size Edge Cover. For all these problem we give polynomial time $f(\text{OPT})$ -approximation algorithms for some small function f (the largest approximation ratio we give is OPT^2). Our results indicate a potential separation between the classes W[1] and W[2]; since no W[2]-hard problem is known to have a polynomial time $f(\text{OPT})$ -approximation for any function f . Finally, we answer a question by Marx [25] by showing the well-studied Strongly Connected Steiner Subgraph problem (which is W[1]-hard and does not have any polynomial time constant factor approximation) has a constant factor FPT-approximation.

1 Introduction

Parameterized Complexity is a two-dimensional generalization of “P vs. NP” where in addition to the overall input size n , one studies the effects on the computational complexity of a secondary measurement that captures additional relevant information. This additional information can be, for example, a structural restriction on the input distribution considered, such as a bound on the treewidth of an input graph or the

*Department of Computer Science, University of Maryland at College Park, USA. Supported in part by NSF CAREER award 1053605, NSF grant CCF-11161626, ONR YIP award N000141110662, DARPA/AFOSR grant FA9550-12-1-0423, and a University of Maryland Research and Scholarship Award (RASA). The first author is also supported by a Simons Award for Graduate Students in Theoretical Computer Science. The second author is also with AT&T Labs. Email: {rchitnis, hajiagha}@cs.umd.edu

[†]Rutgers University, Camden, NJ. Supported in part by NSF grant number 434923. Email: guyk@camden.rutgers.edu

size of a solution. For general background on the theory see [10]. For decision problems with input size n , and a parameter k , the two dimensional analogue (or generalization) of P, is solvability within a time bound of $O(f(k)n^{O(1)})$, where f is a function of k alone. Problems having such an algorithm are said to be *fixed parameter tractable* (FPT). The W -hierarchy is a collection of computational complexity classes: we omit the technical definitions here. The following relation is known amongst the classes in the W -hierarchy: $FPT = W[0] \subseteq W[1] \subseteq W[2] \subseteq \dots$. It is widely believed that $FPT \neq W[1]$, and hence if a problem is hard for the class $W[i]$ (for any $i \geq 1$) then it is considered to be fixed-parameter intractable. We say that a problem is W -hard if it is hard for the class $W[i]$ for some $i \geq 1$. When the parameter is the size of the solution then the most famous examples of $W[1]$ -hard and $W[2]$ -hard problems are Clique and Set Cover respectively. We define these two problems below:

Clique

Input : An undirected graph $G = (V, E)$, and an integer k

Problem: Does G have a clique of size at least k ?

Parameter: k

Set Cover

Input: Universe $U = \{u_1, u_2, \dots, u_n\}$ and a collection $\mathcal{S} = \{S_1, S_2, \dots, S_m\}$ of subsets of U such that $\bigcup_{j=1}^m S_j = U$.

Problem: Is there a subcollection $\mathcal{S}' \subseteq \mathcal{S}$ such that $|\mathcal{S}'| \leq k$ and $\bigcup_{S_i \in \mathcal{S}'} S_i = U$?

Parameter: k

The next natural question is whether these fixed-parameter intractable problems at least admit parameterized approximation algorithms.

1.1 Parameterized Approximation Algorithms

We follow the notation from Marx [26]. Any NP-optimization problem can be described as $O = (I, \text{sol}, \text{cost}, \text{goal})$, where I is the set of instances, $\text{sol}(x)$ is the set of feasible solutions for instance x , the positive integer $\text{cost}(x; y)$ is the cost of solution y for instance x , and goal is either min or max. We assume that $\text{cost}(x, y)$ can be computed in polynomial time, $y \in \text{sol}(x)$ can be decided in polynomial time, and $|y| = |x|^{O(1)}$ holds for every such y .

Definition 1.1. Let $\rho : \mathbb{N} \rightarrow \mathbb{R}_{\geq 1}$ be a computable function such that $\rho(k) \geq 1$ for every $k \geq 1$; if $\text{goal} = \text{min}$ then $k \cdot \rho(k)$ is nondecreasing and if the $\text{goal} = \text{max}$ then $k/\rho(k)$ is unbounded and nondecreasing. An **FPT approximation algorithm** with approximation ratio ρ for O is an algorithm \mathbb{A} that, given an input $(x, k) \in \Sigma^* \times \mathbb{N}$ satisfying $\text{sol}(x) \neq \emptyset$ and

$$\begin{cases} \text{opt}(x) \leq k & \text{if goal} = \text{min} \\ \text{opt}(x) \geq k & \text{if goal} = \text{max} \end{cases} \quad (*)$$

computes $y \in \text{sol}(x)$ such that

$$\begin{cases} \text{cost}(x, y) \leq k \cdot \rho(k) & \text{if goal} = \text{min} \\ \text{cost}(x, y) \geq k/\rho(k) & \text{if goal} = \text{max} \end{cases} \quad (**)$$

We require that on input (x, k) the algorithm \mathbb{A} runs in $f(k) \cdot |x|^{O(1)}$ time, for some computable function f .

Note that if the input does not satisfy (*), then the output can be arbitrary.

Remark 1.2. Given an output $y \in \text{sol}(x)$ we can check in FPT time if it satisfies (**). Hence we can assume that an FPT approximation algorithm always¹ either outputs a $y \in \text{sol}(x)$ that satisfies (**) or outputs a default value reject. We call such an FPT approximation algorithm that has this property as **normalised**.

Classic polynomial-time approximation algorithms determine the performance ratio by comparing the output with the optimum. In FPT approximation algorithms there is a subtle difference: we compare the output to the parameter to determine the approximation ratio. Fellows [14] asked about finding an FPT approximation algorithm for W[2]-hard Dominating Set (which is a special case of Set Cover), or ruling out such a possibility. The following conjecture due to Marx (personal communication) is widely believed in the FPT community:

Conjecture 1.3. Both Set Cover and Clique do not admit an FPT algorithm with approximation ratio ρ , for any function ρ .

However to the best of our knowledge there has been no progress towards proving this conjecture, even under assumptions from complexity theory. In this paper we take a first step towards proving Conjecture 1.3, under well-known and reasonable² assumptions from complexity theory like the Exponential Time Hypothesis (ETH) of Impagliazzo et al. [19] and the Projection Games Conjecture (PGC) of Moshkovitz [29].

For both minimization and maximization problems, the most interesting and practical case is the input (x, k) when $k = \text{OPT}(x)$. This motivates the definition of the following variant of FPT approximation algorithms:

Definition 1.4. Let $\rho : \mathbb{N} \rightarrow \mathbb{R}_{\geq 1}$ be a computable function such that $\rho(k) \geq 1$ for every $k \geq 1$; if goal=min then $k \cdot \rho(k)$ is nondecreasing and if goal=max then $k/\rho(k)$ is unbounded and nondecreasing. An **FPT optimum approximation algorithm** for O with approximation ratio ρ is an algorithm \mathbb{A}' that, given an input $x \in \Sigma^*$ satisfying $\text{sol}(x) \neq \emptyset$ outputs a $y \in \text{sol}(x)$ such that

$$\begin{cases} \text{cost}(x, y) \leq \text{OPT}(x) \cdot \rho(\text{OPT}(x)) & \text{if goal} = \min \\ \text{cost}(x, y) \geq \text{OPT}(x) / \rho(\text{OPT}(x)) & \text{if goal} = \max \end{cases} \quad (1)$$

We require that on input x the algorithm \mathbb{A} runs in $f(\text{OPT}(x)) \cdot |x|^{O(1)}$ time, for some computable function f .

In Section 2.2, we show the following theorem:

Theorem 1.5. Let O be a minimization problem in NP and \mathbb{A} be an FPT approximation algorithm for O with ratio ρ . On input (x, k) let the running time of \mathbb{A} be $f(k) \cdot |x|^{O(1)}$ for some non-decreasing computable function f . Then O also has an FPT optimum approximation algorithm \mathbb{A}' with approximation ratio ρ , and whose running time on input x is also $f(\text{OPT}(x)) \cdot |x|^{O(1)}$

Hence for minimization problems, it is enough to prove hardness results only for the notion of FPT optimum approximation algorithms (see Definition 1.4). We do not know any relation between the two definitions for maximization problems, and hence we prove hardness results for both Definition 1.1 and Definition 1.4.

¹even if the input does not satisfy (*)

²It is very important to only work under well-believed assumptions, since otherwise we will be able to prove pretty much what we want, but it is of no value.

2 Our Results

We make the first progress towards proving Conjecture 1.3, under standard assumptions from complexity theory. In particular for Set Cover we assume the Exponential Time Hypothesis (ETH) [19] and the Projection Games Conjecture (PGC) [29]³. The PGC gives a reduction from SAT to Projection Games. Composing this with the standard reduction from Projection Games to Set Cover gives a reduction from SAT to Set Cover. Since the ETH gives a lower bound on the running time of SAT, we are able to show the following inapproximability result in Section 4:

Theorem 2.1. *Under the ETH and PGC,*

1. *There exist constants $F_1, F_2 > 0$ such that the Set Cover problem does not admit an FPT **optimum** approximation algorithm with ratio $\rho(OPT) = OPT^{F_1}$ in $2^{OPT^{F_2}} \cdot \text{poly}(N, M)$ time, where N is the size of the universe and M is the number of sets.*
2. *There exist constants $F_3, F_4 > 0$ such that the Set Cover problem does not admit an FPT approximation algorithm with ratio $\rho(k) = k^{F_3}$ in $2^{k^{F_4}} \cdot \text{poly}(N, M)$ time, where N is the size of the universe and M is the number of sets.*

In Section 5, we consider the Clique problem. We use the result of Zuckerman [32] which states that it is NP-hard to get an $O(n^{1-\epsilon})$ -approximation for Clique. Given any problem $X \in \text{NP}$, by using the Zuckerman reduction from X to Clique allows us to show the following result.

Theorem 2.2. *Unless $\text{NP} \subseteq \text{SUBEXP}$, for every $1 > \delta > 0$*

1. *There exists a constant $F(\delta) > 0$ such that Clique has no FPT **optimum** approximation with ratio $\rho(OPT) = OPT^{1-\delta}$ in $2^{OPT^F} \cdot \text{poly}(n)$ time, where n is the number of vertices in the graph.*
2. *There exists a constant $F'(\delta) > 0$ such that Clique has no FPT approximation with ratio $\rho(k) = k^{1-\delta}$ in $2^{k^{F'}} \cdot \text{poly}(n)$ time, where n is the number of vertices in the graph.*

2.1 Polytime $f(OPT)$ -approximation for W-hard problems

We also deal with the following question: given that a problem is W-hard, can we maybe get a good polynomial-time approximation for the problem? Any problem in NP can be solved in $2^{n^{O(1)}}$ time by simply enumerating all candidates for the witness. If the parameter k is at least $\log n$, then we immediately have $2^k \geq n$ and the problem can be solved in $2^{n^{O(1)}} \leq 2^{k^{O(1)}}$ time which is FPT time in k . So for large values of the parameter the brute force algorithm itself becomes an FPT algorithm. Hence the intrinsic hardness to obtain FPT algorithms for intractable problems is when the parameter k is small (say at most $\log n$). In this case, we show how to replace the impossible FPT solution by a good approximation, namely $f(OPT)$ approximation for some small function f . We systematically design polynomial-time $f(OPT)$ approximation algorithms for a number of W[1]-hard minimization problems such as Minimum Size Edge Cover, Strongly Connected Steiner Subgraph, Directed Steiner Forest and Directed Steiner Network. Each of the aforementioned problems is known to have strong inapproximability (in terms of input size). Since we can assume OPT is small, this implies $f(OPT)$ is small as well. Therefore for these W[1]-hard problems, if the parameter is large then we can get an FPT algorithm, otherwise if the parameter is small (then OPT is small as well, otherwise we can reject for these minimization problems) and we obtain a reasonable approximation in polynomial time. These results point towards a separation between the classes W[1] and W[2] since we do not know any W[2]-hard problem which has a polynomial-time $f(OPT)$ -approximation, for any function

³The PGC is stated in Section 3.2, and the ETH is stated in Section 3.1

f . In fact, Marx (personal communication) conjectured that the W[2]-hard Set Cover problem does not have a polynomial-time $f(OPT)$ -approximation for any function f .

Finally in Section 7 we show that the well-studied W[1]-hard Strongly Connected Steiner Subgraph problem has an FPT 2-approximation algorithm. This answers a question by Marx [25] regarding finding a problem which is fixed-parameter intractable, does not have a constant factor approximation in polynomial time but admits a constant factor FPT approximation. To the best of our knowledge no such W[2]-hard problem (parameterized by solution size) is known, and this indicates another potential difference between W[1] and W[2].

2.2 Proof of Theorem 1.5

Let $x \in \Sigma^*$ be the input for \mathbb{A}' . The algorithm \mathbb{A}' runs the algorithm \mathbb{A} on the instances $(x, 1), (x, 2), \dots$ until the first k such that the output of \mathbb{A} on (x, k) is a solution of cost at most $k \cdot \rho(k)$. Then \mathbb{A}' outputs $\mathbb{A}(x, k)$. By Definition 1.1, we know that $k \leq OPT(x)$. Hence $k \cdot \rho(k) \leq OPT(x) \cdot \rho(OPT(x))$. It remains to analyze the running time of \mathbb{A}' .

Since $k \leq OPT(x)$, the running time of \mathbb{A}' is upper bounded by $\sum_{i=1}^k f(i) \cdot |x|^{O(1)} \leq \sum_{i=1}^{OPT(x)} f(i) \cdot |x|^{O(1)} = \left(\sum_{i=1}^{OPT(x)} f(i)\right) \cdot |x|^{O(1)} \leq OPT(x) \cdot f(OPT(x)) \cdot |x|^{O(1)} = f(OPT(x)) \cdot |x|^{O(1)}$, since f is non-decreasing and $OPT(x) \leq |x|$. \square

3 Conjectures from Computational Complexity

In this section, we describe two conjectures from computational complexity that we work with in this paper.

3.1 Exponential Time Hypothesis

Impagliazzo, Paturi and Zane [19] formulated the following conjecture which is known as the Exponential Time Hypothesis (ETH).

Exponential Time Hypothesis (ETH)
 3-SAT cannot be solved in $2^{o(n)}$ time where n is the number of variables.

Using the Sparsification Lemma of Calabro, Impagliazzo and Paturi [5], the following lemma was shown in [19].

Lemma 3.1. *Assuming ETH, 3-SAT cannot be solved in $2^{o(m)}$ time where m is the number of clauses.*

In the reductions from 3-SAT to Clique, Vertex Cover and Independent Set, the number of vertices formed in the graphs is equal to the number of clauses in the 3-SAT instance and hence Lemma 3.1 gives evidence against subexponential algorithms for the above three problems. This is enough to give some belief in the ETH. We note that ETH and its variants have been used to prove lower bounds in both FPT [22] and exact exponential algorithms [8]. We refer to [23] for a nice survey on lower bounds using ETH. In this paper, we use ETH to give inapproximability results for Set Cover.⁴

⁴It is not clear what Moshkovitz [29] refers to as the size of a SAT instance. If it is the number of variables, then we use the ETH as is. Otherwise if it refers to the number of clauses, then we are still fine by the Sparsification Lemma [5]

3.2 The Projection Games Conjecture

First we define a *projection game*. Note that with a loss of factor two we can assume that the alphabet is the same for both sides. The input to a projection game consists of:

- A bipartite graph $G = (V_1, V_2, E)$
- A finite alphabets Σ
- Constraints (also called projections) given by $\pi_e : \Sigma \rightarrow \Sigma$ for every $e \in E$.

The goal is to find an assignment $\phi : V_1 \cup V_2 \rightarrow \Sigma$ that *satisfies* as many of the edges as possible. We say that an edge $e = \{a, b\} \in E$ is satisfied, if the projection constraint holds, i.e., $\pi_e(\phi(a)) = \phi(b)$. We denote the size of a projection game by $n = |V_1| + |V_2| + |E|$.

Conjecture 3.2. (Projection Games Conjecture [29]) *There exists $c > 0$ such that for every $\varepsilon > 0$, there is a polynomial reduction $RED-1$ from SAT^5 to Projection Games which maps an instance I of SAT to an instance I_1 of Projection Games such that:*

1. *If a YES instance I of SAT satisfies $|I|^c \geq \frac{1}{\varepsilon}$, then all edges of I_1 can be satisfied.*
2. *If a NO instance I of SAT satisfies $|I|^c \geq \frac{1}{\varepsilon}$, then at most ε -fraction of the edges of I_1 can be satisfied.*
3. *The size of I_1 is almost-linear in the size of I , and is given by $|I_1| = n = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$.*
4. *The alphabet Σ for I_1 has size $\text{poly}(\frac{1}{\varepsilon})$.*

A weaker version of the conjecture is known, but the difference is that the alphabet in [30] has size $\exp(\frac{1}{\varepsilon})$. As pointed out in [29], the Projection Games Conjecture is an instantiation of the Sliding Scale Conjecture of Bellare et al. [2] from 1993. Thus, in fact this conjecture is actually 20 years old. But we have reached a state of knowledge now that it seems likely that the Projection Games Conjecture will be proved not long from now (see Section 1.2 of [29]). Thus it seems that posing this conjecture is quite reasonable. In contrast to this is the Unique Games Conjecture [20]. On the positive side, it seems that the Unique Games Conjecture is much more influential than the Projection Games Conjecture. But it seems unlikely (to us) that the Unique Games Conjecture will be resolved in the near future.

4 An FPT Inapproximability Result for Set Cover

The goal of this section is to prove Theorem 2.1.

4.1 Reduction from Projection Games to Set Cover

The following reduction from Projection Games to Set Cover is known, see [1, 24]. We sketch a proof below for completeness.

Theorem 4.1. *There is a reduction $RED-2$ from Projection Games to Set Cover which maps an instance $I_1 = (G = (V_1, V_2, E), \Sigma, \pi)$ of Projection Games to an instance I_2 of Set Cover such that:*

1. *If all edges of I_1 can be satisfied, then I_2 has a set cover of size $|V_1| + |V_2|$.*
2. *If at most ε -fraction of edges of I_1 can be satisfied, then the size of a minimum set cover for I_2 is at least $\frac{|V_1| + |V_2|}{\sqrt{32\varepsilon}}$*

⁵SAT is the standard Boolean satisfiability problem

3. The instance I_2 has $|\Sigma| \times (|V_1| + |V_2|)$ sets and the size of the universe is $2^{O(\frac{1}{\sqrt{\varepsilon}})} \times |\Sigma|^2 \times |E|$
4. The time taken for the reduction is upper bounded by $2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(|\Sigma|) \times \text{poly}(|E| + |V_1| + |V_2|)$

4.1.1 Proof of Theorem 4.1

Definition 4.2. An (m, ℓ) -set system consists of a universe B and collection of subsets $\{C_1, \dots, C_m\}$ such that if the union of any sub-collection of ℓ -sets from the collection $\{C_1, \dots, C_m, \overline{C_1}, \dots, \overline{C_m}\}$ is B , then the collection must contain both C_i and $\overline{C_i}$ for some $i \in [m]$.

It is known that an (m, ℓ) -set system with a universe size $|B| = O(2^{2\ell} m^2)$ exists, and can be constructed in $2^{O(\ell)} \cdot m^{O(1)}$ time [1]. Consider the following reduction:

Projection Games Instance:

$(G = (V_1, V_2, E), \Sigma, \pi)$ such that $|\Sigma| = m$.

Set Cover Instance:

Let B be a (m, ℓ) set system. The universe for the set cover instance consists of $E \times B$. Define the following subsets of $E \times B$

- For all vertices $v \in V_2$ and $x \in \Sigma$, define the subset $S_{v,x} = \bigcup_{e \ni v} \{e\} \times C_x$
- For all vertices $u \in V_1$ and $y \in \Sigma$, define the subset $S_{u,y} = \bigcup_{e \ni u} \{e\} \times \overline{C_{\pi_e(y)}}$

The Set Cover instance produced is $(E \times B, \{S_{w,x} \mid w \in V_1 \cup V_2, x \in \Sigma\})$

The following theorem is shown in [21]. We give a proof below for the sake of completeness.

Theorem 4.3. *If all edges of G can be satisfied then the instance of Set Cover constructed has a set cover of size $|V_1| + |V_2|$. On the other hand if at most $\frac{2}{\ell}$ -fraction of edges of G can be satisfied then the minimum size of set cover for the Set Cover instance constructed above is $\frac{\ell}{8}(|V_1| + |V_2|)$.*

Assuming Theorem 4.3, we obtain Theorem 4.1 by setting $\varepsilon = \frac{2}{\ell}$ in Theorem 4.3. Recall that $m = |\Sigma|$. Hence the size of the universe is $|E \times B| = |E| \times |B| = |E| \times 2^{O(\frac{1}{\sqrt{\varepsilon}})} \times |\Sigma|^2$ and the number of sets is $|\Sigma| \times (|V_1| + |V_2|)$.

We prove Theorem 4.3 via the following two lemmas:

Lemma 4.4. *If all the edges of G can be satisfied then the instance of Set Cover has a set cover of size $|V_1| + |V_2|$*

Proof. Let $\delta : V_1 \cup V_2 \rightarrow \Sigma$ denote a labeling for G that satisfies all the edges E . Pick the following set of sets $\mathcal{S} = S_{w, \delta(w)} \mid w \in V_1 \cup V_2$. The number of sets in \mathcal{S} is $|V_1| + |V_2|$. We claim that \mathcal{S} is a valid set cover for $E \times B$. For every edge $e = (u, v)$ we show the following holds

$$\{e\} \times B \subseteq S_{u, \delta(u)} \cup S_{v, \delta(v)} \quad (2)$$

The definition of $S_{u, \delta(u)}$ and $S_{v, \delta(v)}$ implies

$$\{e\} \times C_{\delta(v)} \subseteq S_{v, \delta(v)} \text{ and } \{e\} \times \overline{C_{\Pi_e(\delta(u))}} \subseteq S_{u, \delta(u)} \quad (3)$$

Since δ satisfies all the edges (and hence also satisfies e), we have $\Pi_e(\delta(u)) = \delta(v)$. Therefore

$$\{e\} \times \overline{C_{\delta(v)}} = \{e\} \times \overline{C_{\Pi_e(\delta(u))}} \subseteq S_{u, \delta(u)} \quad (4)$$

Now we can see that Equation 3 and Equation 4 imply Equation 2. Moreover, taking the union of the containment relation implied by Equation 2 for all edges e , we get $E \times B \subseteq \bigcup_{u \in V_1 \cup V_2} S_{u, \delta(u)}$ which completes the proof. \square

Lemma 4.5. *If at most $\frac{2}{\ell}$ -fraction of edges of G can be satisfied then the minimum size of set cover for the Set Cover instance is $\frac{\ell}{8}(|V_1| + |V_2|)$.*

Proof. We prove the contrapositive. Suppose there is a set cover \mathcal{S} with $|\mathcal{S}| < \frac{\ell}{8}(|V_1| + |V_2|)$. Then for each vertex w define the set of labels

$$L_w = \{c \in \Sigma \mid S_{w,c} \in \mathcal{S}\}$$

This implies that $|\mathcal{S}| = \sum_{w \in V_1 \cup V_2} |L_w|$. Hence the average cardinality of L_w satisfies

$$\frac{\sum_{w \in V_1 \cup V_2} |L_w|}{|V_1| + |V_2|} = \frac{|\mathcal{S}|}{|V_1| + |V_2|} \leq \frac{\ell}{8}$$

If there are more than $\frac{\ell}{4}$ vertices such that $|L_w| > \frac{\ell}{2}$, then the total sum $\sum_w |L_w|$ would be greater than $\frac{\ell}{8}$, which is a contradiction. Hence at least $\frac{3}{4}$ of the vertices $w \in V_1 \cup V_2$ satisfy $|L_w| \leq \frac{\ell}{2}$. Since we can assume that the bipartite graph of the Projection Games instance is regular, we have that at least half the edges have both endpoints (u, v) such that $|L_u| < \frac{\ell}{2}$ and $|L_v| < \frac{\ell}{2}$.

Definition 4.6. *We say that an edge $e = (u, v)$ is frugally covered if $|L_u| < \frac{\ell}{2}$ and $|L_v| < \frac{\ell}{2}$.*

Consider the following labeling δ' for G : for every $w \in V_1 \cup V_2$ choose an element from L_w uniformly at random. We now show that the expected fraction of edges covered by δ' is at least $\frac{2}{\ell^2}$, which will complete the proof.

To show this, we obtain that each *frugally covered* edge is satisfied by δ' with probability at least $\frac{4}{\ell^2}$. Since there are at least $\frac{|E|}{2}$ *frugally covered* edges, we are done. It remains to show that any frugally covered edge is satisfied by δ' with probability at least $\frac{4}{\ell^2}$. Let $e = (u, v)$ be any frugally covered edge. Let $L_u = \{a_1, a_2, \dots, a_p\}$ and $L_v = \{b_1, b_2, \dots, b_q\}$. Since e is frugally covered we have $\frac{\ell}{2} > \max\{p, q\}$. The sets in \mathcal{S} completely cover $E \times B$, and hence they also cover $e \times B$. Note, that for any vertex $w \notin \{u, v\}$ we have $|S_{w,x} \cap \{eB\}| = 0$ for all $x \in \Sigma$. In other words, no element of the set $e \times B$ can be covered by any of the sets $S_{w,x}$ for any vertex $w \notin \{u, v\}$. Therefore the set $e \times B$ is covered by the sets chosen for vertices u and v . That is,

$$\{e\} \times B \subseteq \left(\bigcup_{i=1}^p S_{u,a_i} \right) \cup \left(\bigcup_{j=1}^q S_{v,b_j} \right)$$

By definition of $S_{v,x}$, we have $S_{v,x} \cap (\{e\} \times B) = \{e\} \times C_x$. Similarly $S_{u,y} \cap (\{e\} \times B) = \{e\} \times \bar{C}_{\Pi_e(y)}$. Restricting the sets S_{u,a_i} and S_{v,b_j} to $\{e\} \times B$ in the above containment we get

$$\{e\} \times B \subseteq \left(\{e\} \times \bigcup_{i=1}^p \bar{C}_{\Pi(a_i)} \right) \cup \left(\{e\} \times \bigcup_{j=1}^q C_{b_j} \right)$$

Therefore, we have

$$B \subseteq \left(\bigcup_{i=1}^p \bar{C}_{\Pi(a_i)} \right) \cup \left(\bigcup_{j=1}^q C_{b_j} \right)$$

This means that B is covered by a family of $p + q \leq \ell$ sets, all of which are either of the form C_i or \bar{C}_i . Since (B, C_i) form a (m, ℓ) set system, there exists an index i such that both C_i and \bar{C}_i are present among the $p + q$ sets. This implies for some a_i, b_j we have $\Pi_e(a_i) = b_j$. Since we choose the labels uniformly at random, with probability $\frac{1}{pq}$ we choose both $\delta'(u) = a_i$ and $\delta'(v) = b_j$. Thus the probability that e is satisfied by δ' is at least $\frac{1}{pq} \geq \left(\frac{2}{\ell}\right)^2 = \frac{4}{\ell^2}$. \square

4.2 Composing the Two Reductions:

Composing the reductions from Conjecture 3.2 and Theorem 4.1 we get:

Theorem 4.7. *There exists $c > 0$, such that for every $\varepsilon > 0$ there is a reduction RED-3 from SAT to Set Cover which maps an instance I of SAT to an instance I_2 of Set Cover such that*

1. *If a YES instance I of SAT satisfies $|I|^c \geq \frac{1}{\varepsilon}$, then I_2 has a set cover of size β .*
2. *If a NO instance I of SAT satisfies $|I|^c \geq \frac{1}{\varepsilon}$, then I_2 does not have a set cover of size less than $\frac{\beta}{\sqrt{32\varepsilon}}$.*
3. *The size N of the universe for the instance I' is $2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|)$.*
4. *The number M of sets for the set cover instance I' is $\text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|)$.*
5. *The total time required for RED-3 is $\text{emph}(|I|) + 2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|)$.*

where $\beta \leq |I_1| = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$. Note that the number of elements is very large compared to the number of sets.

Proof. We apply the reduction from Theorem 4.1 with $|\Sigma| = \text{poly}(\frac{1}{\varepsilon})$ and $|V_1| + |V_2| + |E| = n = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$. Substituting these values in Conjecture 3.2 and Theorem 4.1, we get the parameters as described in the given theorem. We work out each of the values below:

1. If I is a YES instance of SAT satisfying $\varepsilon \geq \frac{1}{|I|^c}$, then RED-1 maps it to an instance $I_1 = (G = (V_1, V_2, E), \Sigma, \pi)$ of Projection Games such that all edges of I_1 can be satisfied. Then RED-2 maps I_1 to an instance I_2 of Set Cover such that I_2 has a set cover of size $\beta = |V_1| + |V_2| \leq |V_1| + |V_2| + |E| = |I_1| = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$.
2. If I is a NO instance of SAT satisfying $\varepsilon \geq \frac{1}{|I|^c}$, then RED-1 maps it to an instance $I_1 = (G = (V_1, V_2, E), \Sigma, \pi)$ of Projection Games such that at most ε -fraction of the edges of I_1 can be satisfied. Then RED-2 maps I_1 to an instance I_2 of Set Cover such that I_2 does not have a set cover of size $\frac{\beta}{\sqrt{32\varepsilon}}$, where β is as calculated above.
3. By Theorem 4.1(3), the size of the universe is $2^{O(\frac{1}{\sqrt{\varepsilon}})} \times |\Sigma|^2 \times |E|$. Observing that $|\Sigma| = \text{poly}(\frac{1}{\varepsilon})$ and $|E| \leq |I_1| = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$, it follows that the size of the universe is $2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|)$.
4. By Theorem 4.1(3), the number of sets is $|\Sigma| \times (|V_1| + |V_2|)$. Observing that $|\Sigma| = \text{poly}(\frac{1}{\varepsilon})$ and $|V_1| + |V_2| \leq |I_1| = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$, it follows that the number of sets is $\text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|)$.
5. Since RED-3 is the composition of RED-1 and RED-2, the time required for RED-3 is the summation of the times required for RED-1 and RED-2. By Conjecture 3.2, the time required for RED-1 is $\text{poly}(|I|)$. By Theorem 4.1(4), the time required for RED-2 is at most $2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(|\Sigma|) \times \text{poly}(|E| + |V_1| + |V_2|)$. Observing that $|\Sigma| = \text{poly}(\frac{1}{\varepsilon})$ and $|V_1| + |V_2| + |E| = |I_1| = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$, it follows that the time required for RED-2 is at most $2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|)$. Adding up the two, the time required for RED-3 is at most $\text{poly}(|I|) + 2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|)$.

□

Finally we are ready to prove Theorem 2.1.

4.3 Proof of Theorem 2.1(1)

The roadmap of the proof is as follows: suppose to the contrary there exists an FPT **optimum** approximation algorithm, say \mathbb{A} , for Set Cover with ratio $\rho(OPT) = OPT^{F_1}$ in $2^{OPT^{F_2}} \cdot \text{poly}(N, M)$ time, where N is the size of the universe and M is the number of sets (recall Definition 1.4). We will choose the constant F_1 such that using RED-3 from Theorem 4.7 (which assumes PGC), the algorithm \mathbb{A} applied to the instance I_2 will be able to decide the instance I_1 of SAT. Then to violate ETH we will choose the constant F_2 such that the running time of \mathbb{A} summed up with the time required for RED-3 is subexponential in $|I|$.

Let $c > 0$ be the constant from Conjecture 3.2. Fix some constant $1 > \delta > 0$ and let $c^* = \min\{c, 2 - 2\delta\}$. Note that $\frac{c^*}{2} \leq 1 - \delta$. Choosing $\varepsilon = \frac{1}{|I|^{c^*}}$ implies $\varepsilon \geq \frac{1}{|I|^c}$, since $c \geq c^*$. We carry out the reduction RED-3 given by Theorem 4.7. From Conjecture 3.2(3), we know that $|I_1| = |I|^{1+o(1)} \cdot \text{poly}(\frac{1}{\varepsilon})$. Let $\lambda > 0$ be a constant such that the $\text{poly}(\frac{1}{\varepsilon})$ is upper bounded by $(\frac{1}{\varepsilon})^\lambda$. Then Theorem 4.7 implies $\beta \leq |I|^{1+o(1)} \cdot (\frac{1}{\varepsilon})^\lambda$. However we have chosen $\varepsilon = \frac{1}{|I|^{c^*}}$, and hence asymptotically we get

$$\beta \leq |I|^2 \cdot |I|^{\lambda c^*} = |I|^{2+\lambda c^*} \quad (5)$$

Choose the constant F_1 such that $\frac{c^*}{4(2+\lambda c^*)} \geq F_1$. Suppose Set Cover has an FPT optimum approximation algorithm \mathbb{A} with ratio $\rho(OPT) = OPT^{F_1}$ (recall Definition 1.4). We show that this algorithm \mathbb{A} can decide the SAT problem. Consider an instance I of SAT, and let $I_2 = \text{RED-3}(I)$ be the corresponding instance of Set Cover. Run the FPT approximation algorithm on I_G , and let $\mathbb{A}(I_2)$ denote the output of ALG. We have the following two cases:

- $\frac{\beta}{\sqrt{32\varepsilon}} \leq \mathbb{A}(I_2)$: Then we claim that I is a NO instance of SAT. Suppose to the contrary that I is a YES instance of SAT. Then Theorem 4.7(1) implies $\beta \geq OPT(I_2)$. Hence $\frac{\beta}{\sqrt{32\varepsilon}} \leq \mathbb{A}(I_2) \leq OPT \cdot \rho(OPT) = OPT^{1+F_1} = \beta^{1+F_1} \Rightarrow \frac{1}{\sqrt{32\varepsilon}} \leq \beta^{F_1}$. However, asymptotically we have $\frac{1}{\sqrt{32\varepsilon}} = \frac{|I|^{\frac{c^*}{2}}}{\sqrt{32}} > |I|^{\frac{c^*}{4}} \geq (|I|^{2+\lambda c^*})^{F_1} = \beta^{F_1}$, where the last two inequalities follows from the choice of F_1 and Equation 5 respectively. This leads to a contradiction, and therefore I is a NO instance of SAT.
- $\frac{\beta}{\sqrt{32\varepsilon}} > \mathbb{A}(I_2)$: Then we claim that I is a YES instance of SAT. Suppose to the contrary that I is a NO instance of SAT. Then Theorem 4.7(2) implies $OPT(I_2) \geq \frac{\beta}{\sqrt{32\varepsilon}}$. Therefore we have $\frac{\beta}{\sqrt{32\varepsilon}} > \mathbb{A}(I_2) \geq OPT(I_2) \geq \frac{\beta}{\sqrt{32\varepsilon}}$.

Therefore we run the algorithm \mathbb{A} on the instance I_2 and compare the output $\frac{\beta}{\sqrt{32\varepsilon}}$ with n^ε . As seen above, this comparison allows us to decide the SAT problem.

We now choose the constant F_2 such that the running time of \mathbb{A} summed up with the time required for RED-3 is subexponential in $|I|$.

By Theorem 4.7(5), the total time taken by RED-3 is $\text{poly}(|I|) + 2^{O(\frac{1}{\sqrt{\varepsilon}})} \times \text{poly}(\frac{1}{\varepsilon}) \times \text{poly}(|I|) = \text{poly}(|I|) + 2^{O(|I|^{\frac{c^*}{2}})} \times \text{poly}(|I|^{\frac{c^*}{2}}) \times \text{poly}(|I|) = \text{poly}(|I|) + 2^{o(I)} \cdot \text{poly}(|I|)$ since $\frac{c^*}{2} \leq 1 - \delta$. Hence total time taken by RED-3 is subexponential in I . We now show that there exists a constant F_2 such that the claimed running time of $2^{OPT^{F_2}} \cdot \text{poly}(N, M)$ for the algorithm \mathbb{A} is subexponential in $|I|$, thus contradicting ETH. We do not have to worry about the $\text{poly}(N, M)$ factor: the reduction time is subexponential in $|I|$, and hence $\max\{N, M\}$ is also upper bounded by a subexponential function of $|I|$. Hence, we essentially want to choose a constant $F_2 > 0$ such that $OPT^{F_2} \leq M^{F_2} = o(|I|)$. From Theorem 4.7(4), we know that $M \leq |\Sigma| \times |V_1 + V_2|$. Since $|\Sigma| = \text{poly}(\frac{1}{\varepsilon})$, let $\alpha > 0$ be a constant such that the $|\Sigma| \leq (\frac{1}{\varepsilon})^\alpha$. We have seen earlier in the proof that

$|V_1 + V_2| \leq |I_1| \leq |I|^2 \cdot (\frac{1}{\varepsilon})^\lambda = |I|^{2+c^*\lambda}$. Therefore $M^{F_2} \leq (|I|^{2+c^*\lambda+c^*\alpha})^{F_2}$. Choosing $F_2 < \frac{1}{2+\lambda c^*+2\alpha c^*}$ gives $\text{OPT}^{F_2} = o(|I|)$, which is what we wanted to show. \square

4.4 Proof of Theorem 2.1(2)

Observe that due to Theorem 1.5, Theorem 2.1(1) implies Theorem 2.1(2).

5 An FPT Inapproximability Result for Clique

We use the following theorem due to Zuckerman [32], which in turn is a derandomization of a result of Hastad [18].

Theorem 5.1. [18, 32] *Let X be any problem in NP. For any constant $\varepsilon > 0$ there exists a polynomial time reduction from X to Clique so that the gap between the clique sizes corresponding to the YES and NO instances of X is at least $n^{1-\varepsilon}$, where n is the number of vertices of the Clique instances.*

5.1 Proof of Theorem 2.2(1)

Fix a constant $1 > \delta > 0$. Set $0 < \varepsilon = \frac{\delta}{\delta+2}$, or equivalently $\delta = \frac{2\varepsilon}{1-\varepsilon}$. Let X be any problem in NP. Let the Hastad-Zuckerman reduction from X to Clique [18, 32] which creates a gap of at least $n^{1-\varepsilon}$ map an instance I of X to the corresponding instance I_G of Clique. Since the reduction is polynomial, we know that $n = |I_G| = |I|^D$ for some constant $D(\varepsilon) > 0$. Note that D depends on ε , which in turn depends on δ . Hence, ultimately D depends on δ . If I is a YES instance of X , then I_G contains a clique of size at least $n^{1-\varepsilon}$ since each graph has a trivial clique of size one and the gap between YES and NO instances of Clique is at least $n^{1-\varepsilon}$. Similarly, observe that a graph on n vertices can have a clique of size at most n . To maintain the gap of at least $n^{1-\varepsilon}$, it follows if I is a NO instance of X then the maximum size of a clique in I_G is at most n^ε . To summarize, we have

- If I is a YES instance, then $\text{OPT}(I_G) \geq n^{1-\varepsilon}$
- If I is a NO instance, then $\text{OPT}(I_G) \leq n^\varepsilon$

Suppose Clique has an FPT **optimum** approximation algorithm \mathbb{A} with ratio $\rho(\text{OPT}) = \text{OPT}^{1-\delta}$ (recall Definition 1.4). We show that this algorithm \mathbb{A} can decide the problem X . Consider an instance I of X , and let I_G be the corresponding instance of Clique. Run the FPT approximation algorithm on I_G , and let $\mathbb{A}(I_G)$ denote the output of \mathbb{A} . We have the following two cases:

- $n^\varepsilon \geq \mathbb{A}(I_G)$: Then we claim that I is a NO instance of X . Suppose to the contrary that I is a YES instance of X , then we have $n^\varepsilon \geq \mathbb{A}(I_G) \geq \frac{\text{OPT}_{I_G}}{\rho(\text{OPT}(I_G))} = (\text{OPT}(I_G))^\delta \geq (n^{1-\varepsilon})^\delta = n^{2\varepsilon}$, which is a contradiction.
- $n^\varepsilon < \mathbb{A}(I_G)$: Then we claim that I is a YES instance of X . Suppose to the contrary that I is a NO instance of X , then we have $n^\varepsilon < \mathbb{A}(I_G) \leq \text{OPT}(I_G) \leq n^\varepsilon$, which is a contradiction.

We run the algorithm \mathbb{A} on the instance I_G and compare the output $\mathbb{A}(I_G)$ with n^ε . As seen above, this comparison allows us to decide the problem X . We now show how to choose the constant F such that the running $2^{\text{OPT}^F} \cdot \text{poly}(n)$ is subexponential in $|I|$. We claim that $F = \frac{1}{D+1}$ works. Note that $\text{OPT}(I_G) \leq n$ always. Hence $2^{\text{OPT}^F} \cdot \text{poly}(n) \leq 2^{n^F} \cdot \text{poly}(n) = 2^{(|I|^D)^F} \cdot \text{poly}(|I|^D) = 2^{|I|^{DF}} \cdot \text{poly}(|I|) = 2^{|I|^{\frac{D}{D+1}}} \cdot \text{poly}(|I|) = 2^{o(I)} \cdot \text{poly}(|I|)$. This implies we can solve X in subexponential time using \mathbb{A} . However X was any problem chosen from the class NP, and hence $\text{NP} \subseteq \text{SUBEXP}$. \square

	W[1]-hardness	Polytime Approx. Ratio
Strongly Connected Steiner Forest	Guo et al. [15]	OPT^ε (Lemma 6.1)
Directed Steiner Forest	Lemma 6.2	$OPT^{1+\varepsilon}$ (Lemma 6.3)
Directed Steiner Network	Lemma 6.4	OPT^2 (Lemma 6.5)
Minimum Edge Cover	Lemma 6.6	$OPT - 1$ (Lemma 6.7)
Directed Multicut	Marx and Razgon [28]	$3 \cdot OPT$ [16]

Figure 1: Polytime $f(OPT)$ -approximation for W[1]-hard problems

5.2 Proof of Theorem 2.2(2)

Fix a constant $1 > \delta > 0$. Set $0 < \varepsilon = \frac{\delta}{\delta+1}$, or equivalently $\delta = \frac{\varepsilon}{1-\varepsilon}$. Let X be any problem in NP. Let the Hastad-Zuckerman reduction from X to Clique [18, 32] which creates a gap of at least $n^{1-\varepsilon}$ map an instance I of X to the corresponding instance I_G of Clique. Since the reduction is polynomial, we know that $n = |I_G| = |I|^D$ for some constant $D(\varepsilon) > 0$. Note that D depends on ε , which in turn depends on δ . Hence, ultimately D depends on δ . If I is a YES instance of X , then I_G contains a clique of size at least $n^{1-\varepsilon}$ since each graph has a trivial clique of size one and the gap between YES and NO instances of Clique is at least $n^{1-\varepsilon}$. Similarly, observe that a graph on n vertices can have a clique of size at most n . To maintain the gap of at least $n^{1-\varepsilon}$, it follows if I is a NO instance of X then the maximum size of a clique in I_G is at most n^ε .

Suppose Clique has an FPT approximation algorithm ALG with ratio $\rho(k) = k^{1-\delta}$ (recall Definition 1.1). We show that this algorithm ALG can decide the problem X . Set $k = n^\varepsilon$. On the input (I_G, n^ε) to ALG, there are two possible outputs:

- ALG outputs reject $\Rightarrow OPT(I_G) < n^\varepsilon \Rightarrow I$ is a NO instance of X
- ALG outputs a clique of size $\geq \frac{k}{\rho(k)} \Rightarrow OPT(I_G) \geq \frac{k}{\rho(k)} = \frac{k}{k^{1-\delta}} = k^\delta = (n^\varepsilon)^\delta = n^{1-\varepsilon} \Rightarrow I$ is a YES instance of X

Therefore the FPT approximation algorithm ALG can decide the problem $X \in \text{NP}$.

We now show how to choose the constant F' such that the running $\exp(k^{F'}) \cdot \text{poly}(n)$ is subexponential in $|I|$. We claim that $F' = \frac{1}{\varepsilon \cdot D + 1}$ works. This is because $2^{k^{F'}} \cdot \text{poly}(n) = 2^{n^{\varepsilon F'}} \cdot \text{poly}(n) = 2^{|I|^{\varepsilon D F'}} \cdot \text{poly}(|I|^D) = 2^{|I|^{\frac{\varepsilon D}{\varepsilon D + 1}}} \cdot \text{poly}(|I|) = 2^{o(|I|)} \cdot \text{poly}(|I|)$.

This implies we can solve X in subexponential time using ALG. However X was any problem chosen from the class NP, and hence $\text{NP} \subseteq \text{SUBEXP}$. \square

6 Polytime $f(OPT)$ -approximation for W[1]-hard problems

In Section 2.1 we have seen the motivation for designing polynomial time $f(OPT)$ -approximation algorithms for W[1]-hard problems such as Minimum Size Edge Cover, Strongly Connected Steiner Subgraph, Directed Steiner Forest and Directed Steiner Network. Our results are summarized in Figure 1.

6.1 The Strongly Connected Steiner Subgraph Problem

Lemma 6.1. *For any constant $\varepsilon > 0$, the Strongly Connected Steiner Subgraph problem has a $2 \cdot OPT^\varepsilon$ -approximation in polynomial time.*

Proof. Fix any constant $\varepsilon > 0$. Let G_{rev} denote the reverse graph obtained from G , i.e., reverse the orientation of each edge. Any solution of the Strongly Connected Steiner Subgraph instance must contain

a path from t_1 to each terminal in $T \setminus t_1$ and vice versa. Consider the following two instances of the Directed Steiner Tree problem: $I_1 = (G, t_1, T \setminus t_1)$ and $I_2 = (G_{rev}, t_1, T \setminus t_1)$. In [6] an $|T|^\varepsilon$ -approximation is designed for Directed Steiner Tree in polynomial time, for any constant $\varepsilon > 0$. Let E_1, E_2 be the $|T|^\varepsilon$ -approximate solutions for the two instances and say that their optimum solutions are OPT_1, OPT_2 respectively. Let OPT be the size of optimum solution for the Strongly Connected Steiner Subgraph instance, then clearly $|OPT| \geq \max\{|OPT_1|, |OPT_2|\}$. Clearly $E_1 \cup E_2$ is a solution for the Strongly Connected Steiner Subgraph instance as E_j is a solution for I_j for $1 \leq j \leq 2$. It now remains to bound the size of this solution: $|E_1 \cup E_2| \leq |E_1| + |E_2| \leq |T|^\varepsilon |OPT_1| + |T|^\varepsilon |OPT_2| = |T|^\varepsilon (|OPT_1| + |OPT_2|) \leq 2|T|^\varepsilon |OPT|$. As every terminal has at least one incoming edge (and these edges are pairwise disjoint) we get that $OPT \geq |T| = k$. Therefore $|T|^\varepsilon \leq OPT^\varepsilon$ which implies a $2 \cdot OPT^\varepsilon$ -approximation factor. \square

6.2 The Directed Steiner Forest Problem

The Directed Steiner Forest problem is LabelCover hard and thus admits no $2^{\log^{1-\varepsilon} n}$ -approximation for any constant ε [9]. The best know approximation factor for the problem is $n^{\frac{2}{3}}$ [12, 3]. We now define the problem formally:

Directed Steiner Forest

Input : A digraph $G = (V, E)$ and a set of terminals $T = \{(s_1, t_1), \dots, (s_k, t_k)\}$.

Problem: Does there exist a set $E' \subseteq E$ such that $|E'| \leq p$ and (V, E') has a $s_i \rightarrow t_i$ path for every $i \in [k]$.

Parameter: p

Lemma 6.2. *The Directed Steiner Forest problem is $W[1]$ -hard parameterized by solution size plus number of terminal pairs.*

Proof. We give a reduction from the Strongly Connected Steiner Subgraph problem. Consider an instance (G, T, p) of Strongly Connected Steiner Subgraph where $T = \{t_1, t_2, \dots, t_\ell\}$. We now build a new graph G^* as follows:

- Add 2ℓ new vertices: for every $i \in \ell$, we introduce vertices r_i and s_i .
- For every $i \in [\ell]$, add the edges (r_i, t_i) and (t_i, s_i) .

Let the terminal pairs be $T^* = \{(r_i, s_j) \mid 1 \leq i, j \leq \ell; i \neq j\}$. We claim that the Strongly Connected Steiner Subgraph instance (G, T) has a solution of size p if and only if there is a solution for the Directed Steiner Forest instance (G^*, T^*) of size $p + 2\ell$.

Suppose there is a solution for the Strongly Connected Steiner Subgraph instance of size p . Adding the edges from $E(G^*) \setminus E(G)$ clearly gives a solution for the Directed Steiner Forest instance of size $p + 2\ell$. Conversely, suppose we have a solution for the Directed Steiner Forest instance of size $p + 2\ell$. Since t_i is the only out-neighbor, in-neighbor of r_i, s_i respectively the solution must contain all the edges from $E(G^*) \setminus E(G)$. Removing these edges clearly gives a solution of size p to the Strongly Connected Steiner Subgraph instance. Note that $|T^*| = \ell(\ell - 1)$. Since Strongly Connected Steiner Subgraph is $W[1]$ -hard parameterized by solution size plus number of terminals, we have that Directed Steiner Forest is $W[1]$ -hard parameterized by solution size plus number of terminal pairs. \square

Lemma 6.3. *The Directed Steiner Forest problem admits an $OPT^{1+\varepsilon}$ -approximation in polynomial time.*

Proof. Let $S = \{v \mid \exists x \text{ such that } (v, x) \in T\}$. For every $v \in S$, let $T_v = \{x \mid (v, x) \in T\}$. For each $v \in S$, let the optimum for the instance (G, v, T_v) of Directed Steiner Tree be OPT_v . Clearly $OPT_v \leq OPT$, where OPT is

the optimum of the given Directed Steiner Forest instance. For each $v \in S$, we take the $|T_v|^\epsilon$ -approximation given in [6] for the instance (G, v, T_v) of Directed Steiner Tree, and output the union of all these Steiner trees. Clearly this gives a feasible solution. We now analyze the cost.

Since each vertex in S must have its own outgoing edge in the solution, we have $|S| \leq OPT$. Similarly, for every $v \in S$ each vertex of T_v must have its own incoming edge in the solution, and hence $|T_v| \leq OPT$. Hence the cost of our solution is upper bounded by $\sum_{v \in S} |T_v|^\epsilon \cdot OPT_v \leq \sum_{v \in S} OPT^\epsilon \cdot OPT \leq |S| \cdot OPT^{1+\epsilon} \leq OPT^{2+\epsilon}$. Therefore, we get a $OPT^{1+\epsilon}$ -approximation. \square

6.3 The Directed Steiner Network Problem

The Directed Steiner Network problem is not known to admit any non-trivial approximation and of course is LabelCover hard. We define the problem formally:

Directed Steiner Network

Input : A digraph $G = (V, E)$, a set of terminals $T = \{(s_1, t_1), \dots, (s_k, t_k)\}$, a demand d_i between s_i, t_i for every $i \in [k]$

Problem: Does there exist a set $E' \subseteq E$ such that $|E'| \leq p$ and (V, E') has d_i disjoint $s_i \rightarrow t_i$ paths for every $i \in [k]$.

Parameter: p

Lemma 6.4. *The Directed Steiner Network problem is W[1]-hard parameterized by solution size plus number of terminal pairs.*

Proof. The lemma follows from Lemma 6.2 and the fact that the Directed Steiner Forest problem is a special case of the Directed Steiner Network problem with $d_i = 1$ for every i . \square

Lemma 6.5. *The Directed Steiner Network problem admits an OPT^2 -approximation in polynomial time.*

Proof. Let $S = \{v \mid \exists x \text{ such that } (v, x) \in T\}$ and $S' = \{x \mid \exists v \text{ such that } (v, x) \in T\}$. For every $v \in S$, let $T_v = \{x \mid (v, x) \in T\}$. For each $v \in S$ and $x \in T_v$, let the demand for the pair (v, x) be d_{vx} . Make v as a sink and x as a source. Using min-cost max-flow, find the smallest edge set, say E_{vx} , such that there are d_{vx} disjoint $v \rightarrow x$ paths in G . Clearly $|E_{vx}| \leq OPT$ since any solution for the Directed Steiner Network instance must contain d_{vx} disjoint $v \rightarrow x$ paths. Also, as seen before $OPT \geq |S|$ since each vertex in S must have at least one outgoing edge in any solution. Similarly $OPT \geq |S'|$. We output $\bigcup_{v \in S, x \in T_v} E_{vx}$ as our solution. Clearly this is a feasible solution. Its cost is $\sum_{v \in S, x \in T_v} |E_{vx}| \leq \sum_{v \in S, x \in T_v} OPT \leq |S| \cdot |S'| \cdot OPT \leq OPT \cdot OPT \cdot OPT$, and hence this gives an OPT^2 -approximation. \square

6.4 The Minimum Size Edge Cover Problem

In this section, we show that the Minimum Size Edge Cover problem is W[1]-hard parameterized by size of the solution, and it admits an $(OPT - 1)$ -approximation in polynomial time. The best approximation for Minimum Size Edge Cover is $O(n^{0.172})$ due to Chlamtac et al. [7].

Minimum Size Edge Cover

Input : A graph $G = (V, E)$ and an integer k

Problem : Does there exist a set $S \subseteq V$ such that $|S| \leq p$ and the number of edges with both endpoints in S is at least k .

Parameter : p

To show that W[1]-hardness of Minimum Size Edge Cover we reduce from the Multicolored Clique problem which is known to be W[1]-hard [13]⁶:

Lemma 6.6. *The Minimum Size Edge Cover problem is W[1]-hard parameterized by the size of the solution.*

Proof. Given an instance $I_1 = (G, \phi, p)$ of Multicolored Clique, we can consider another instance $I_2 = (G, k, p)$ of Minimum Size Edge Cover where $k = \binom{p}{2}$. Clearly if I_1 is a YES instance, then G has a multi-colored clique and I_2 is a YES instance. In the other direction, if I_2 is a YES instance then the p -sized set must form a clique in G , and must be in different color classes as ϕ is a proper vertex coloring. This shows that Minimum Size Edge Cover is W[1]-hard parameterized by the size of the covering set. \square \square

Now we show give an approximation algorithm for the Minimum Size Edge Cover problem.

Lemma 6.7. *The Minimum Size Edge Cover problem admits an $(OPT - 1)$ -approximation in polynomial time.*

Proof. Let k be the desired number of edges in the solution and let OPT be the minimum number of vertices required. If there is a feasible solution, then there must be at least k edges in the graph. Pick any k edges, and let p' the size of the set which is the union of their endpoints. Clearly $p' \leq 2k$. Since $k \leq \frac{OPT(OPT-1)}{2}$, we have $p' \leq 2k \leq OPT(OPT - 1)$, and hence we get a $(OPT - 1)$ -approximation⁷. \square

7 Constant Factor FPT Approximation For SCSS

In this section we show that SCSS has an FPT 2-approximation. We define the problem formally:

Strongly Connected Steiner Subgraph (SCSS)

Input : An directed graph $G = (V, E)$, a set of terminals $T = \{t_1, t_2, \dots, t_\ell\}$ and an integer p

Problem: Does there exists a set $E' \subseteq E$ such that $|E'| \leq p$ and the graph $G' = (V, E')$ has a $t_i \rightarrow t_j$ path for every $i \neq j$

Parameter: p

Lemma 7.1. *Strongly Connected Steiner Subgraph has an FPT 2-approximation.*

Proof. Let G_{rev} denote the reverse graph obtained from G , i.e., reverse the orientation of each edge. Any solution of SCSS instance must contain a path from t_1 to each terminal in $T \setminus t_1$ and vice versa. Consider the following two instances of the Directed Steiner Tree problem: $I_1 = (G, t_1, T \setminus t_1)$ and $I_2 = (G_{rev}, t_1, T \setminus t_1)$, and let their optimum be OPT_1, OPT_2 respectively. Let OPT be the optimum of given SCSS instance and k be the parameter. If $OPT > k$ then we output anything (see Definition 1.1). Otherwise we have $k \geq OPT \geq \max\{OPT_1, OPT_2\}$. We know that the Directed Steiner Tree problem is FPT parameterized by the size of the solution [11]. Hence we find the values OPT_1, OPT_2 in time which is FPT in k . Clearly the union of solutions for I_1 and I_2 as a solution for instance I of SCSS. The final observation is $OPT_1 + OPT_2 \leq OPT + OPT = 2 \cdot OPT$. \square

Guo et al. [15] show that SCSS is W[1]-hard parameterized by solution size plus number of terminals. It is known that SCSS has no $\log^{2-\epsilon} n$ -approximation in polynomial time for any fixed $\epsilon > 0$, unless NP has quasi-polynomial Las Vegas algorithms [17]. Combining these facts with Lemma 7.1 implies that SCSS

⁶Cai [4] has independently shown the W[1]-hardness of Minimum Size Edge Cover with parameter p . They call this problem as MAXIMUM p -VERTEX SUBGRAPH

⁷In fact, any feasible solution gives a $(OPT - 1)$ -approximation

is a W[1]-hard problem that is not known to admit a constant factor approximation in polynomial time but has a constant factor FPT approximation. This answers a question by Marx [25]. Previously the only such problem known was a variant of the Almost-2-SAT problem [31] called 2-ASAT-BFL, due to Marx and Razgon [27].

References

- [1] S. Arora and C. Lund. *Approximation algorithms for NP-hard problems*. Ed. D Hochbaum, PWS Publishing Co., Boston, MA, USA, 1997.
- [2] M. Bellare, S. Goldwasser, C. Lund, and A. Russell. Efficient probabilistically checkable proofs and applications to approximations. In *STOC*, 1993.
- [3] P. Berman, A. Bhattacharyya, K. Makarychev, S. Raskhodnikova, and G. Yaroslavtsev. Improved approximation for the directed spanner problem. In *ICALP (1)*, pages 1–12, 2011.
- [4] L. Cai. Parameterized complexity of cardinality constrained optimization problems. *Comput. J.*, 51(1):102–121, 2008.
- [5] C. Calabro, R. Impagliazzo, and R. Paturi. A duality between clause width and clause density for sat. In *IEEE Conference on Computational Complexity*, pages 252–260, 2006.
- [6] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha, and M. Li. Approximation algorithms for directed Steiner problems. *J. of Algorithms*, 33, 1999.
- [7] E. Chlamtac, M. Dinitz, and R. Krauthgamer. Everywhere-sparse spanners via dense subgraphs. *FOCS*, 2012.
- [8] M. Cygan, H. Dell, D. Lokshantov, D. Marx, J. Nederlof, Y. Okamoto, R. Paturi, S. Saurabh, and M. Wahlström. On Problems as Hard as CNFSAT. *To appear in CCC 2012*.
- [9] Y. Dodis and S. Khanna. Design networks with bounded pairwise distance. In *STOC*, 1999.
- [10] R. Downey and M. Fellows. *Parameterized Complexity*. Springer-Verlag, 1999.
- [11] S. E. Dreyfus and R. A. Wagner. The steiner problem in graphs. *Networks*, 1(3), 1971.
- [12] M. Feldman, G. Kortsarz, and Z. Nutov. Improved approximation algorithms for directed steiner forest. *J. Comput. Syst. Sci.*, 78(1):279–292, 2012.
- [13] M. Fellows, D. Hermelin, F. Rosamond, and S. Vialette. On the parameterized complexity of multiple-interval graph problems. *Theor. Comp. Sci.*, 410(1), 2009.
- [14] M. R. Fellows, J. Guo, D. Marx, and S. Saurabh. Data Reduction and Problem Kernels (Dagstuhl Seminar 12241). *Dagstuhl Reports*, 2(6):26–50, 2012.
- [15] J. Guo, R. Niedermeier, and O. Suchý. Parameterized complexity of arc-weighted directed steiner problems. *SIAM J. Discrete Math.*, 25(2):583–599, 2011.
- [16] A. Gupta. Improved results for directed multicut. In *SODA*, pages 454–455, 2003.
- [17] E. Halperin and R. Krauthgamer. Polylogarithmic inapproximability. In *STOC '03: Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 585–594, New York, NY, USA, 2003. ACM Press.

- [18] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. In *FOCS*, 1996.
- [19] R. Impagliazzo and R. Paturi. On the complexity of k-sat. *J. Comput. Syst. Sci.*, 62(2), 2001.
- [20] S. Khot. On the Unique Games Conjecture. In *FOCS*, page 3, 2005.
- [21] Lecture-Notes. <http://www.cs.washington.edu/education/courses/cse533/05au/lecture14.pdf>.
- [22] D. Lokshtanov, D. Marx, and S. Saurabh. Known algorithms on graphs on bounded treewidth are probably optimal. In *SODA*, pages 777–789, 2011.
- [23] D. Lokshtanov, D. Marx, and S. Saurabh. Lower bounds based on the exponential time hypothesis. *Bulletin of the EATCS*, 105:41–72, 2011.
- [24] C. Lund and M. Yannakakis. On the hardness of approximating minimization problems. *J. ACM*, 41(5):960–981, 1994.
- [25] D. Marx. Parameterized complexity and approximation algorithms. *Comput. J.*, 51(1), 2008.
- [26] D. Marx. Completely inapproximable monotone and antimonotone parameterized problems. *Journal of Computer and System Sciences*, 79(1), 2013.
- [27] D. Marx and I. Razgon. Constant ratio fixed-parameter approximation of the edge multicut problem. *Inf. Process. Lett.*, 109(20):1161–1166, 2009.
- [28] D. Marx and I. Razgon. Fixed-parameter tractability of multicut parameterized by the size of the cutset. In *STOC*, pages 469–478, 2011.
- [29] D. Moshkovitz. The Projection Games Conjecture and The NP-Hardness of In n-Approximating Set-Cover. *APPROX*, 2012.
- [30] D. Moshkovitz and R. Raz. Two-query PCP with subconstant error. *J. ACM*, 57(5), 2010.
- [31] I. Razgon and B. O’Sullivan. Almost 2-sat is fixed-parameter tractable. *J. Comput. Syst. Sci.*, 75(8):435–450, 2009.
- [32] D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In *STOC*, pages 681–690, 2006.