



University
of Glasgow

Jerrum, M., and Meeks, K. (2014) *The parameterised complexity of counting connected subgraphs and graph motifs*. Journal of Computer and System Sciences . ISSN 0022-0000

Copyright © 2014 Elsevier Inc.

A copy can be downloaded for personal non-commercial research or study, without prior permission or charge

Content must not be changed in any way or reproduced in any format or medium without the formal permission of the copyright holder(s)

When referring to this work, full bibliographic details must be given

<http://eprints.gla.ac.uk/100035/>

Deposited on: 08 December 2014

Enlighten – Research publications by members of the University of Glasgow
<http://eprints.gla.ac.uk>

The Parameterised Complexity of Counting Connected Subgraphs and Graph Motifs ^{*}

Mark Jerrum[†] and Kitty Meeks^{‡§}

School of Mathematical Sciences, Queen Mary University of London

Abstract

We introduce a family of parameterised counting problems on graphs, p -#INDUCED SUBGRAPH WITH PROPERTY(Φ), which generalises a number of problems which have previously been studied. This paper focusses on the case in which Φ defines a family of graphs whose edge-minimal elements all have bounded treewidth; this includes the special case in which Φ describes the property of being connected. We show that exactly counting the number of connected induced k -vertex subgraphs in an n -vertex graph is #W[1]-hard, but on the other hand there exists an FPTRAS for the problem; more generally, we show that there exists an FPTRAS for p -#INDUCED SUBGRAPH WITH PROPERTY(Φ) whenever Φ is monotone and all the minimal graphs satisfying Φ have bounded treewidth. We then apply these results to a counting version of the GRAPH MOTIF problem.

1 Introduction

Parameterised counting problems were introduced by Flum and Grohe in [13] and also independently by McCartin [21]. In this paper we focus on problems of the following form:

Input: An n -vertex graph $G = (V, E)$, and $k \in \mathbb{N}$.

Parameter: k .

Question: How many (labelled) k -vertex subsets of V induce graphs with a given property?

It should be noted that, while the statement of this problem is concerned with *induced* subgraphs, it also encompasses problems more often formulated in terms of counting subgraphs that are not necessarily induced. For example, to count the number of k -vertex paths in G (not necessarily induced), we would

^{*}Research supported by EPSRC grant “Computational Counting”

[†]m.jerrum@qmul.ac.uk

[‡]k.meeks@glasgow.ac.uk

[§]Present address: School of Mathematics and Statistics, University of Glasgow

consider the labelled subgraph induced by v_1, \dots, v_k to have the desired property if and only if $v_i v_{i+1}$ is an edge for $1 \leq i \leq k-1$, regardless of what other edges may be present (and then divide the result by two, as this will count each path exactly twice).

Many problems of this form are known to be $\#W[1]$ -hard (see Section 1.2 for definitions of concepts from parameterised complexity), and thus are unlikely to be solvable exactly in time $f(k)n^{O(1)}$ for any function f . A number of these $\#W[1]$ -hard problems are in fact *induced* subgraph counting problems: Chen and Flum [4] demonstrated that problems of counting k -vertex induced paths and of counting k -vertex induced cycles are both $\#W[1]$ -complete, and more generally Chen, Thurley and Weyer [5] showed that it is $\#W[1]$ -complete to count the number of induced subgraphs isomorphic to a given graph from the class \mathcal{C} (p - $\#INDUCED\ SUBGRAPH\ ISOMORPHISM(\mathcal{C})$) whenever \mathcal{C} contains arbitrarily large graphs. Most other subgraph counting problems previously studied in the literature can be described in the following way, for appropriate choices of a class of graphs \mathcal{H} :

p - $\#SUB(\mathcal{H})$

Input: A graph G and an element $H \in \mathcal{H}$.

Parameter: $k = |V(H)|$.

Question: How many subgraphs (not necessarily induced) of G are isomorphic to H ?

Examples of $\#W[1]$ -hard problems of this form include counting the number of k -vertex cliques (p - $\#CLIQUE$ [13]), paths (p - $\#PATH$ [13]), cycles (p - $\#CYCLE$ [13]) and matchings (p - $\#MATCHING$ [6]). Very recently, Curticapean and Marx [7] proved a dichotomy result for p - $\#SUB(\mathcal{H})$, demonstrating that the problem is $\#W[1]$ -complete unless all the graphs in \mathcal{H} have vertex-cover number bounded by some fixed constant, in which case the problem is fixed parameter tractable.

A natural question, therefore, is whether such counting problems, which are hard to solve exactly, can be efficiently approximated. It is shown in [2] that there exists an efficient approximation scheme for p - $\#SUB(\mathcal{H})$ whenever \mathcal{H} is a class of graphs having bounded treewidth,

In Section 1.3.1 below, we introduce formally a family of parameterised counting problems which includes all the specific problems discussed above. This family also includes the problem of counting the number of k -vertex connected induced subgraphs, a problem which we show to be $\#W[1]$ -hard in Section 2. In Section 3 we generalise the approximation result from [2], showing that there exists an FPTRAS for the more general problem of counting the number of (labelled) k -vertex subsets of a graph satisfying a monotone property Φ , provided that the edge-minimal graphs satisfying Φ all have bounded treewidth. Examples of problems in this class for which there exists an FPTRAS include those of counting the number of k -vertex induced subgraphs that are connected, the number of k -vertex induced subgraphs that are Hamiltonian, and the number of k -vertex induced subgraphs that are not bipartite. This last example contrasts with the result of Khot and Raman [19] that deciding whether a graph contains

an induced k -vertex subgraph that is bipartite is W[1]-hard.

Finally, in Section 4, we apply some of these results to a counting version of the problem GRAPH MOTIF, introduced by Lacroix, Fernandes and Sagot [20] in the context of metabolic networks. The problem takes as input an n -vertex coloured graph, together with a *motif* or multiset of colours M , and a solution is a subset U of $|M|$ vertices such that the subgraph induced by U is connected and the colour-(multi)set of U is exactly M . A counting version of this problem was studied by Guillemot and Sikora [15]; we define and analyse a different natural counting version of GRAPH MOTIF, which is a more direct translation of the standard decision version into the counting world.

In the remainder of this section, we first introduce some notation in Section 1.1, then introduce some key concepts in the study of parameterised counting complexity in Section 1.2, before giving formal definitions of the problems we consider in Section 1.3.

1.1 Notation

Given a graph $G = (V, E)$, and a subset $U \subset V$, we write $G[U]$ for the subgraph of G induced by the vertices of U . We denote by \overline{G} the *complement* of G , that is, $\overline{G} = (V, E')$ where $E' = V^{(2)} \setminus E$. If $v \in V$, then $\Gamma(v)$ denotes the set of neighbours of v in G . For any $k \in \mathbb{N}$, we write $[k]$ as shorthand for $\{1, \dots, k\}$, and denote by S_k the set of all permutations on $[k]$, that is, injective functions from $[k]$ to $[k]$. We write $V^{(k)}$ for the set of all subsets of V of size exactly k , and $V^{\underline{k}}$ for the set of k -tuples $(v_1, \dots, v_k) \in V^k$ such that v_1, \dots, v_k are all distinct.

If G is coloured by some colouring $\omega : V \rightarrow [k]$, we say that a subset $U \subset V$ is *colourful* (under ω) if, for every $i \in [k]$, there exists a unique vertex $u \in U$ such that $\omega(u) = i$; note that this can only be achieved if $U \in V^{(k)}$. We write $\omega|_U$ for the restriction of ω to the set U ; if U is colourful under ω then $\omega|_U$ is a bijection.

Given graphs G and H , a *embedding* of H in G is an injective mapping $\theta : V(H) \rightarrow V(G)$ such that, for all $uv \in E(H)$, we have $\theta(u)\theta(v) \in E(G)$.

We will be considering labelled graphs, where a labelled graph is a pair (H, π) such that H is a graph and $\pi : [|V(H)|] \rightarrow V(H)$ is a bijection. We write $\mathcal{L}(k)$ for the set of all labelled graphs on the vertex set $[k]$. Given a graph $G = (V, E)$ and a k -tuple of vertices $(v_1, \dots, v_k) \in V^k$, $G[v_1, \dots, v_k]$ denotes the labelled graph (H, π) where $H = G[\{v_1, \dots, v_k\}]$ and $\pi(i) = v_i$ for each $i \in [k]$. We write $(H, \pi) \subseteq (H', \pi')$ if, for all $e = uv \in E(H)$, $\pi'(\pi^{-1}(u))\pi'(\pi^{-1}(v)) \in E(H')$. Given a collection $\mathcal{H} \subseteq \mathcal{L}(k)$ of labelled graphs, we say that a graph $(H, \pi) \in \mathcal{H}$ is an *edge-minimal* element of \mathcal{H} if there is no $(H', \pi') \in \mathcal{H}$ such that $(H', \pi') \subseteq (H, \pi)$ and H' has strictly fewer edges than H .

We say that (T, \mathcal{D}) is a *tree decomposition* of G if T is a tree and $\mathcal{D} = \{\mathcal{D}(t) : t \in V(T)\}$ is a collection of non-empty subsets of $V(G)$ (or *bags*), indexed by the nodes of T , satisfying:

1. $V(G) = \bigcup_{t \in V(T)} \mathcal{D}(t)$,
2. for every $e = uv \in E(G)$, there exists $t \in V(T)$ such that $u, v \in \mathcal{D}(t)$,

3. for every $v \in V(G)$, if $T(v)$ is defined to be the subgraph of T induced by nodes t with $v \in \mathcal{D}(t)$, then $T(v)$ is connected.

The *width* of the tree decomposition (T, \mathcal{D}) is defined to be $\max_{t \in V(T)} |\mathcal{D}(t)| - 1$, and the *treewidth* of G , written $\text{tw}(G)$, is the minimum width over all tree decompositions of G .

1.2 Parameterised counting complexity

In this section, we introduce key notions from parameterised counting complexity, which we will use in the rest of the paper. A parameterised counting problem is a pair (Π, κ) where, for some finite alphabet Σ , $\Pi : \Sigma^* \rightarrow \mathbb{N}_0$ is a function and $\kappa : \Sigma^* \rightarrow \mathbb{N}$ is a parameterisation (a polynomial-time computable mapping). An algorithm A for a parameterised counting problem (Π, κ) is said to be an *fpt-algorithm* if there exists a computable function f and a constant c such that the running time of A on input I is bounded by $f(\kappa(I))|I|^c$. Problems admitting an fpt-algorithm are said to belong to the class FPT.

To understand the complexity of parameterised counting problems, Flum and Grohe [13] introduce two kinds of reductions between such problems.

Definition. Let (Π, κ) and (Π', κ') be parameterised counting problems.

1. An *fpt parsimonious reduction* from (Π, κ) to (Π', κ') is an algorithm that computes, for every instance I of Π , an instance I' of Π' in time $f(\kappa(I)) \cdot |I|^c$ such that $\kappa'(I') \leq g(\kappa(I))$ and

$$\Pi(I) = \Pi'(I')$$

(for computable functions $f, g : \mathbb{N} \rightarrow \mathbb{N}$ and a constant $c \in \mathbb{N}$). In this case we write $(\Pi, \kappa) \leq_{\text{pars}}^{\text{fpt}} (\Pi', \kappa')$.

2. An *fpt Turing reduction* from (Π, κ) to (Π', κ') is an algorithm A with an oracle to Π' such that

- (a) A computes Π ,
- (b) A is an fpt-algorithm with respect to κ , and
- (c) there is a computable function $g : \mathbb{N} \rightarrow \mathbb{N}$ such that for all oracle queries " $\Pi'(I') = ?$ " posed by A on input x we have $\kappa'(I') \leq g(\kappa(I))$.

In this case we write $(\Pi, \kappa) \leq_{\text{T}}^{\text{fpt}} (\Pi', \kappa')$.

Using these notions, Flum and Grohe introduce a hierarchy of parameterised counting complexity classes, $\#\text{W}[t]$, for $t \geq 1$; this is the analogue of the W-hierarchy for parameterised decision problems. In order to define this hierarchy, we need some more notions related to satisfiability problems.

The definition of levels of the hierarchy uses the following problem, where ψ is a first-order formula with a free relation variable of arity s .

$p\text{-}\#\text{WD}_\psi$

Input: A relational structure¹ \mathcal{A} and $k \in \mathbb{N}$.

Parameter: k .

Question: How many relations $S \subseteq A^s$ of cardinality $|S| = k$ are such that $\mathcal{A} \models \psi(S)$ (where A is the universe of \mathcal{A})?

If Ψ is a class of first-order formulas, then $p\text{-}\#\text{WD-}\Psi$ is the class of all problems $p\text{-}\#\text{WD}_\psi$ where $\psi \in \Psi$. The classes of first-order formulas Σ_t and Π_t , for $t \geq 0$, are defined inductively. Both Σ_0 and Π_0 denote the class of quantifier-free formulas, while, for $t \geq 1$, Σ_t is the class of formulas

$$\exists x_1 \dots \exists x_k \psi,$$

where $\psi \in \Pi_{t-1}$, and Π_t is the class of formulas

$$\forall x_1 \dots \forall x_k \psi,$$

where $\psi \in \Sigma_{t-1}$. We are now ready to define the classes $\#W[t]$, for $t \geq 1$.

Definition ([13, 14]). *For $t \geq 1$, $\#W[t]$ is the class of all parameterised counting problems that are fpt parsimonious reducible to $p\text{-}\#\text{WD-}\Pi_t$.*

Unless $\text{FPT}=\text{W}[1]$, there does not exist an algorithm running in time $f(k)n^{O(1)}$ for any problem that is hard for the class $\#W[1]$ under either fpt parsimonious reductions or fpt Turing reductions. In the setting of this paper, a parameterised counting problem will be considered to be intractable if it is $\#W[1]$ -hard with respect to either form of reduction.

In [14], Flum and Grohe also define a counting version of the A-hierarchy for parameterised problems; this turns out to be easier to use for some of our purposes. The definition is in terms of the following model-checking problem, where C is a class of structures and Ψ a class of formulas.

$p\text{-}\#\text{MC}(C, \Psi)$

Input: A structure $\mathcal{A} \in C$ and a formula $\psi \in \Psi$.

Parameter: $|\psi|$.

Question: What is $|\psi(\mathcal{A})|$?

Here, $\psi(\mathcal{A})$ is the set of tuples $(a_1, \dots, a_k) \in A^k$ such that $\psi(a_1, \dots, a_k)$ is true in \mathcal{A} , where k is the number of free variables in ψ , and A the universe of \mathcal{A} . If C is the class of all structures, we write simply $p\text{-}\#\text{MC}(\Psi)$. The counting analogue of the A-hierarchy is then defined as follows.

Definition ([14]). *For all $t \geq 1$, $\#A[t]$ is the class of all parameterised counting problems reducible to $p\text{-}\#\text{MC}(\Pi_{t-1})$ by an fpt parsimonious reduction.*

It is known that the first levels of these two hierarchies for parameterised counting problems coincide:

¹The relational structure \mathcal{A} of vocabulary τ consists of the *universe* A together with an interpretation $R^{\mathcal{A}}$ of every relation R in τ .

Theorem 1.1 ([14]). $\#W[1] = \#A[1]$.

Thus, to prove that a problem belongs to $\#W[1]$ ($=\#A[1]$) it suffices to show that it is reducible, under fpt parsimonious reductions, to $p\text{-}\#\text{MC}(\Pi_0)$.

When considering approximation algorithms for parameterised counting problems, an “efficient” approximation scheme is an FPTRAS, as introduced by Arvind and Raman [2]; this is the analogue of a FPRAS (fully polynomial randomised approximation scheme) in the parameterised setting.

Definition. An FPTRAS for a parameterised counting problem Π with parameter k is a randomised approximation scheme that takes an instance I of Π (with $|I| = n$), and real numbers $\epsilon > 0$ and $0 < \delta < 1$, and in time $f(k) \cdot g(n, 1/\epsilon, \log(1/\delta))$ (where f is any computable function, and g is a polynomial in n , $1/\epsilon$ and $\log(1/\delta)$) outputs a rational number z such that

$$\mathbb{P}[(1 - \epsilon)\Pi(I) \leq z \leq (1 + \epsilon)\Pi(I)] \geq 1 - \delta.$$

1.3 Problems considered

In this section we begin by introducing a general family of parameterised counting problems on graphs, in which the goal is to count k -tuples of vertices that induce subgraphs with particular properties. We then give formal definitions of the problems we will consider in Sections 2 and 3. Subject to appropriate rescaling, our model can be regarded as a generalisation of many problems that involve counting labelled subgraphs, including $p\text{-}\#\text{CYCLE}$ [13], $p\text{-}\#\text{PATH}$ [13], $p\text{-}\#\text{STREMB}(\mathcal{C})$ [5] and $\#k\text{-MATCHING}$ [6]. Induced subgraph problems that are invariant under relabelling of vertices have also been studied in the literature on parameterised counting, including $p\text{-}\#\text{CLIQUE}$ [13], and can be regarded as instances of a sub-family of problems in our model. A more detailed discussion of how problems previously studied in the literature, including $p\text{-}\#\text{SUB}(\mathcal{H})$, can be expressed in the language of this model is given in [17].

1.3.1 The model

Let Φ be a family (ϕ_1, ϕ_2, \dots) of functions $\phi_k : \mathcal{L}(k) \rightarrow \{0, 1\}$, such that the function mapping $k \mapsto \phi_k$ is computable. For any k , we write \mathcal{H}_{ϕ_k} for the set $\{(H, \pi) \in \mathcal{L}(k) : \phi_k(H, \pi) = 1\}$, and set $\mathcal{H}_\Phi = \bigcup_{k \in \mathbb{N}} \mathcal{H}_{\phi_k}$.

We then define the following problem.

$p\text{-}\#\text{INDUCED SUBGRAPH WITH PROPERTY}(\Phi)$

Input: A graph $G = (V, E)$ and $k \in \mathbb{N}$.

Parameter: k .

Question: What is the cardinality of the set $\{(v_1, \dots, v_k) \in V^k :$

$\phi_k(G[v_1, \dots, v_k]) = 1\}$?

Observe that we can equivalently regard this problem as that of counting induced labelled k -vertex subgraphs that belong to \mathcal{H}_Φ . Note that this generalises

the problem $p\text{-\#SUB}(\mathcal{H})$, as the latter problem only permits counting copies (not necessarily induced) of one particular graph H , whereas $p\text{-\#INDUCED SUBGRAPH WITH PROPERTY}(\Phi)$ allows us to count all k -vertex subgraphs having some more complicated property.

We say that Φ is a *monotone* property if, for every k , whenever $\phi_k(H, \pi) = 1$ for some $(H, \pi) \in \mathcal{L}(k)$, and $(H', \pi') \in \mathcal{L}(k)$ with $(H, \pi) \subseteq (H', \pi')$, then we also have $\phi_k(H', \pi') = 1$. We further describe Φ as a *symmetric* property if the value of $\phi_k(H, \pi)$ depends only on the graph H and not on the labelling of the vertices; this corresponds to “unlabelled” graph problems, such as $p\text{-\#CLIQUE}$. We can define a related problem for symmetric properties:

$p\text{-\#INDUCED UNLABELLED SUBGRAPH WITH PROPERTY}(\Phi)$

Input: A graph $G = (V, E)$ and $k \in \mathbb{N}$.

Parameter: k .

Question: What is the cardinality of the set $\{\{v_1, \dots, v_k\} \in V^{(k)} : \phi_k(G[v_1, \dots, v_k]) = 1\}$?

For any symmetric property Φ , the output of $p\text{-\#INDUCED SUBGRAPH WITH PROPERTY}(\Phi)$ is exactly $k!$ times the output of $p\text{-\#INDUCED UNLABELLED SUBGRAPH WITH PROPERTY}(\Phi)$. The unlabelled version is less general than the labelled version, as this only allows us count induced subgraphs having some particular property rather than, for example, all (not necessarily induced) copies of some fixed graph H . As an example, the labelled version can express problems such as $p\text{-\#MATCHING}$, whereas the former would only allow us to count k -vertex induced subgraphs that contain a perfect matching (ignoring the fact that any one k -vertex induced subgraph may contain many perfect matchings).

Note that, for any $\phi_k \in \Phi$, the problem of determining the cardinality of the set $\{\{v_1, \dots, v_k\} \in V^k : \phi_k(G[v_1, \dots, v_k]) = 1\}$ can easily be expressed as an instance of $p\text{-\#MC}(\Pi_0)$; thus, by Theorem 1.1, we obtain the following result:

Proposition 1.2. *For any Φ , the problem $p\text{-\#INDUCED SUBGRAPH WITH PROPERTY}(\Phi)$ belongs to $\#W[1]$. If Φ is symmetric, then the same is true for $p\text{-\#INDUCED UNLABELLED SUBGRAPH WITH PROPERTY}(\Phi)$.*

In order to give an fpt parsimonious reduction from $p\text{-\#INDUCED UNLABELLED SUBGRAPH WITH PROPERTY}(\Phi)$ to $p\text{-\#MC}(\Pi_0)$, we can imitate a technique used in [14, Lemma 14.31], introducing an additional relation in our structure which imposes an order on the elements in order to ensure that each unlabelled subset is counted exactly once.

1.3.2 Problem definitions

In Section 2, we consider the following problem.

$p\text{-\#CONNECTED INDUCED SUBGRAPH}$

Input: A graph $G = (V, E)$ and $k \in \mathbb{N}$.

Parameter: k .

Question: For how many subsets $U \in V^{(k)}$ is $G[U]$ connected?

This problem is clearly symmetric, and can be regarded as a particular case of the general problem p -#INDUCED UNLABELLED SUBGRAPH WITH PROPERTY(Φ) introduced above. Let \mathcal{T}_k be the set of all trees on k vertices with vertices labelled $1, \dots, k$, and then set $\Phi^{\text{conn}} = (\phi_1^{\text{conn}}, \phi_2^{\text{conn}}, \dots)$, with

$$\phi_k^{\text{conn}}(H, \pi) = \bigvee_{T \in \mathcal{T}_k} \bigwedge_{\{j, l\} \in E(T)} \text{adj}_H(\pi(j), \pi(l)), \quad (1)$$

where $\text{adj}_H(u, w) = 1$ if and only if $uw \in E(H)$, and $\text{adj}_H(u, w) = 0$ otherwise. The tuples $(v_1, \dots, v_k) \in V^k$ such that $\phi_k(G[v_1, \dots, v_k]) = 1$ are then exactly the tuples such that $G[\{v_1, \dots, v_k\}]$ is connected.

In Section 3, we consider the more general problem of solving p -#INDUCED SUBGRAPH WITH PROPERTY(Φ), whenever $\Phi = (\phi_1, \phi_2, \dots)$ is a monotone property and there exists a positive integer t such that, for each ϕ_k , all edge-minimal labelled k -vertex graphs (H, π) such that $\phi_k(H, \pi) = 1$ satisfy $\text{tw}(H) \leq t$. Note that p -#CONNECTED INDUCED SUBGRAPH is a special case of this more general problem: the set of edge-minimal labelled k -vertex graphs (H, π) such that $\phi_k^{\text{conn}}(H, \pi) = 1$ is in fact precisely the set of labelled trees on k vertices.

2 p -#CONNECTED INDUCED SUBGRAPH is #W[1]-complete

In this section, we prove the following result.

Theorem 2.1. p -#CONNECTED INDUCED SUBGRAPH is #W[1]-complete under fpt Turing reductions.

We begin in Section 2.1 by noting some background results we will need for the proof, before demonstrating #W[1]-hardness with a series of fpt Turing reductions in Section 2.2.

2.1 Lattices and Möbius functions

In Section 2.2 we will need to consider the lattice formed by partitions of a k -element set, with a partial order given by the refinement relation. A partition of a set X is a set of disjoint subsets X_1, \dots, X_r of X such that $X = X_1 \cup \dots \cup X_r$; the subsets X_1, \dots, X_r are called the *blocks* of the partition. A partition P' *refines* the partition P if every block of P' is contained in some block of P (note that this ordering is the opposite of that more commonly used for partitions). In this section we recall some existing results about lattices on arbitrary posets and also more specifically about partition lattices, which we will use in the proof of Lemma 2.5.

A lattice is a partially ordered set (\mathcal{P}, \leq) satisfying the condition that, for any two elements $x, y \in \mathcal{P}$, both the *meet* and *join* of x and y also belong to \mathcal{P} , where the meet of x and y , written $x \wedge y$, is defined to be the unique element z such that

1. $z \leq x$ and $z \leq y$, and
2. for any w such that $w \leq x$ and $w \leq y$, we have $w \leq z$,

and the join of x and y , $x \vee y$, is correspondingly defined to be the unique element z' such that

1. $x \leq z'$ and $y \leq z'$, and
2. for any w such that $x \leq w$ and $y \leq w$, we have $z' \leq w$.

We denote by $\hat{1}$ and $\hat{0}$ respectively the “top” and “bottom” elements of the lattice (the “top” element is the unique $x \in \mathcal{P}$ such that, for all $y \in \mathcal{P}$, $y \leq x$, and the bottom element is defined symmetrically).

We will make use of the Möbius function μ on a poset, which is defined inductively by

$$\mu(x, y) = \begin{cases} 1 & \text{if } x = y \\ -\sum_{z: x \leq z < y} \mu(x, z) & \text{for } x < y \\ 0 & \text{otherwise.} \end{cases}$$

In Lemma 2.5, we will consider a so-called *meet-matrix* on a partition lattice, and make use of the following lemma (which follows immediately from a result of Haukkanen [16, Cor. 2]).

Lemma 2.2. *Let x_1, \dots, x_n be the elements of a finite lattice (\mathcal{P}, \leq) , let $f : \mathcal{P} \rightarrow \mathbb{C}$ be a function, and let $A = (a_{ij})_{1 \leq i, j \leq n}$ be the matrix given by $a_{ij} = f(x_i \wedge x_j)$. Then*

$$\det(A) = \prod_{i=1}^n \sum_{x_k \leq x_i} f(x_k) \mu(x_k, x_i),$$

where μ is the Möbius function for \mathcal{P} .

To make use of this result in Section 2.2, we will need to be able to calculate certain values of the Möbius function for the special case in which \mathcal{P} is a partition lattice, ordered so that $x \leq y$ whenever y refines x ; in fact, due to our choice of f below, it will suffice to be able to compute $\mu(\hat{0}, x)$ for each partition x . To do this, we will use the following lemma, which is an immediate consequence of two results of Rota [22, Section 3, Prop. 3 and Section 7, Prop. 3].

Lemma 2.3. *Let \mathcal{P} be the lattice of partitions of a set with n elements, where $x \leq y$ if and only if y refines x . If $x \in \mathcal{P}$ is of rank r , then*

$$\mu(\hat{0}, x) = (-1)^r r!$$

where the rank of x is equal to the number of blocks of x minus one.

Since the rank of any element lies in the range $[0, n - 1]$, we obtain the following immediate corollary.

Corollary 2.4. *Let \mathcal{P} be the lattice of partitions of a set with n elements, where $x \leq y$ if and only if y refines x . Then, for all $x \in \mathcal{P}$, $\mu(\hat{0}, x) \neq 0$.*

2.2 The reduction

In this section, we prove Theorem 2.1. The main work in this proof is to give an fpt Turing reduction from p -#MULTICOLOUR INDEPENDENT SET to p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH, where these two problems are defined as follows.

p -#MULTICOLOUR INDEPENDENT SET

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

Parameter: k .

Question: For how many colourful subsets $U \in V^{(k)}$ is U an independent set in G ?

p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH

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

Parameter: k .

Question: For how many colourful subsets $U \in V^{(k)}$ is $G[U]$ connected?

In this reduction we will set up a system of equations, argue that, with an oracle to p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH, we can compute the entries, and show that the system can be solved to give the number of colourful independent sets in our graph. Throughout, we will need to switch between considering colourful subsets of vertices and partitions of $[k]$. Let \mathcal{P}_k be the set of all partitions of the set $[k]$; thus the cardinality of \mathcal{P}_k is precisely the k^{th} Bell number, B_k . We consider these partitions to be partially ordered by the refinement relation, so $P_i \leq P_j$ if P_j refines P_i . Set $P_1 = \hat{0} = \{[k]\}$ and $P_{B_k} = \hat{1} = \{\{1\}, \dots, \{k\}\}$.

Suppose that $G = (V, E)$ is the k -coloured graph in an instance of p -#MULTICOLOUR INDEPENDENT SET. Given a multicolour subset $U \in V^{(k)}$, we set $P(U)$ to be the partition of $[k]$ in which $i, j \in [k]$ belong to the same set of the partition if and only if the vertices of U with colours i and j belong to the same connected component in $G[U]$.

We define a function $f : \mathcal{P}_k \rightarrow \{0, 1\}$ such that, for any partition $P \in \mathcal{P}_k$,

$$f(P) = \begin{cases} 1 & \text{if } P = \{[k]\} \\ 0 & \text{otherwise.} \end{cases}$$

In the following lemma we set up our system of equations, and use results from Section 2.1 to demonstrate that the system can be solved to determine the number of colourful independent sets in our graph.

Lemma 2.5. *Given all values of $\sum_{U \in V^{(k)}} f(P(U) \wedge P')$ for $P' \in \mathcal{P}_k$, we can compute the number of colourful independent sets in G in time $h(k)$, where h is some computable function.*

Proof. For $1 \leq i \leq B_k$, let N_i be the number of subsets $U \in V^{(k)}$ such that $P(U) = P_i$. Since $\mathcal{P}_{B_k} = \{\{1\}, \dots, \{k\}\}$, our goal is then to calculate N_{B_k} .

Let $A = (a_{ij})_{0 \leq i, j \leq B_k}$ be the matrix given by $a_{ij} = f(P_i \wedge P_j)$. We first claim that $A \cdot \mathbf{N} = \mathbf{z}$ where

$$\mathbf{N} = (N_0, \dots, N_{B_k})^T,$$

and

$$\mathbf{z} = (z_1, \dots, z_{B_k})^T$$

with $z_i = \sum_{U \in V^{(k)}} f(P(U) \wedge P_i)$.

To see that this is true, observe that the i^{th} element of $A \cdot \mathbf{N}$ is

$$\begin{aligned} \sum_{j=1}^{B_k} a_{ij} N_j &= \sum_{j=1}^{B_k} f(P_i \wedge P_j) N_j \\ &= \sum_{j=1}^{B_k} \sum_{\substack{U \in V^{(k)} \\ P(U) = P_j}} f(P_i \wedge P_j) \\ &= \sum_{U \in V^{(k)}} f(P_i \wedge P(U)) \\ &= z_i, \end{aligned}$$

as required. Thus it suffices to prove that the matrix A is nonsingular, as then we can compute N_{B_k} as the last element of $A^{-1} \cdot \mathbf{z}$.

To see that this is indeed the case, first note that, by Lemma 2.2,

$$\begin{aligned} \det(A) &= \prod_{j=1}^{B_k} \sum_{P_i \leq P_j} f(P_i) \mu(P_i, P_j) \\ &= \prod_{j=1}^{B_k} \mu(\hat{0}, P_j). \end{aligned}$$

Thus it suffices to verify that all values of $\mu(\hat{0}, P_j)$ for $P_j \in \mathcal{P}$ are non-zero; but this follows immediately from Corollary 2.4. Hence $\det(A) \neq 0$ and A is nonsingular, as required. \square

Now we show that, with an oracle to p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH, we can compute the values required to set up the equations in the previous lemma, completing the reduction from p -#MULTICOLOUR INDEPENDENT SET to p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH.

Lemma 2.6. *There exists a computable function g such that, with an oracle to p -#MULTICOLOUR CONNECTED INDUCED SUBSET, the value of $\sum_{U \in V^{(k)}} f(P(U) \wedge P_i)$ can be computed, for every $P_i \in \mathcal{P}_k$, in time $g(k) \cdot n^{O(1)}$. Moreover, for every oracle call, the parameter value is at most $2k$.*

Proof. We begin by considering how to compute the values of $\sum_{U \in V^{(k)}} f(P(U) \wedge P_i)$ for a single $P_i \in \mathcal{P}_k$. Suppose $P_i = \{X_1, \dots, X_\ell\}$, where each $X_j \subset [k]$.

We construct a new coloured graph G_i , with vertex set $V(G_i) = V(G) \cup \{x_1, \dots, x_\ell\}$, and where the colouring c of $V(G)$ is extended to $V(G_i)$ by setting $c(x_j) = k + j$ for $1 \leq j \leq \ell$. G_i has edge-set

$$E(G_i) = E(G) \cup \bigcup_{1 \leq j \leq \ell} \{x_j v : v \text{ has colour } d \text{ for some } d \in X_j\}.$$

Suppose that $W \subset V(G_i)$ is a multicoloured subset of $V(G_i)$ (so $|W| = k + \ell$, and all vertices in W have distinct colours), and set $U = W \cap V(G)$. Note that, in order for W to be colourful, we must have $W \setminus U = V(G_i) \setminus V(G) = \{x_1, \dots, x_\ell\}$. We make the following claim.

Claim 1. *$G_i[W]$ is connected if and only if $f(P(U) \wedge P_i) = 1$, that is, if and only if the finest partition that is refined by both $P(U)$ and P_i is in fact $\{[k]\}$.*

Proof of claim. Suppose first that $G_i[W]$ is connected. It suffices to prove that, for any $u_1, u_2 \in U$, $c(u_1)$ and $c(u_2)$ belong to the same block of $P(U) \wedge P_i$. Note that, by connectedness of $G_i[W]$, there must exist a path in $G_i[W]$ from u_1 to u_2 . We now proceed by induction on the length of a shortest u_1 - u_2 path in $G_i[W]$.

For the base case, suppose that there is at most one internal vertex on such a path. In this case, either u_1 and u_2 belong to the same connected component of $G[U]$ (in which case we are done, since by definition $c(u_1)$ and $c(u_2)$ then belong to the same block of $P(U)$ and hence $P(U) \wedge P_i$), or else there is a single internal vertex $x_j \in W \setminus U$ lying on this path. Thus $u_1, u_2 \in \Gamma(x_j)$, implying by the construction of G_i that $c(u_1), c(u_2) \in X_j$, so $c(u_1)$ and $c(u_2)$ belong to the same block of P_i and hence of $P(U) \wedge P_i$. This completes the proof of the base case.

We may now assume that there are at least two internal vertices on a shortest u_1 - u_2 path, and that the result holds for any u'_1, u'_2 that are connected by a shorter path in $G_i[W]$. Since there are at least two internal vertices on the shortest u_1 - u_2 path in $G_i[W]$, and no two vertices in $W \setminus U$ are adjacent, there must be some vertex $u_3 \in U \setminus \{u_1, u_2\}$ that lies on this path. But then there exists a shorter u_1 - u_3 path in $G_i[W]$, implying by the inductive hypothesis that $c(u_1)$ and $c(u_3)$ belong to the same block of $P(U) \wedge P_i$. Similarly, we see that $c(u_2)$ and $c(u_3)$ belong to the same block of $P(U) \wedge P_i$, and hence it must be that $c(u_1)$ and $c(u_2)$ belong to the same block of $P(U) \wedge P_i$, as required.

We now consider the reverse implication. Suppose that $G_i[W]$ is not connected, so this graph has connected components with vertex sets W_1, \dots, W_r ,

where $r \geq 2$. We claim that the partition $P_W = \{c(W_1 \cap U), \dots, c(W_r \cap U)\}$ of $[k]$ is refined by both $P(U)$ and P_i ; hence

$$P(U) \wedge P_i \geq P_W > \hat{0},$$

so $P(U) \wedge P_i \neq \hat{0}$, as required. To see that this claim holds, it suffices to check that, for every block X of $P(U)$ or P_i , the vertices having colours from X belong to the same component of $G_i[W]$. If $X \in P(U)$ then this follows immediately, since by definition blocks of $P(U)$ are sets of colours that appear in the same connected component of $G[U]$, and so must certainly belong to the same connected component of $G_i[W]$. Suppose therefore that $X \in P_i$. But then $X = X_j$ for some $1 \leq j \leq \ell$, and so all vertices of U with colours from X are adjacent to x_j , and hence belong to the same component of $G_i[W]$. This completes the proof of the claim.

It therefore follows that $G_i[W]$ is connected if and only if $P(U) \wedge P_i = \hat{0}$, as required. \square (Claim 1)

Thus, by Claim 1, $\sum_{U \in V^{(k)}} f(P(U) \wedge P_i)$ is exactly equal to the number of colourful connected subsets in G_i . Thus, with $B_k < k^k$ calls to an oracle to p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH, we can compute the value of $\sum_{U \in V^{(k)}} f(P(U) \wedge P_i)$ for every $P_i \in \mathcal{P}_k$; for each call, the parameter value $k + \ell$ is at most $2k$. \square

Using Lemmas 2.6 and 2.5, it is now straightforward to prove the main result of this section.

Proof of Theorem 2.1. The fact that p -#CONNECTED INDUCED SUBGRAPH \in #W[1] follows immediately from Proposition 1.2, so it suffices to prove that the problem is #W[1]-hard. To do this, we give a sequence of fpt Turing reductions from p -#CLIQUE, shown to be #W[1]-complete in [13].

p -#CLIQUE $\leq_{\mathbf{T}}^{\text{fpt}}$ p -#MULTICOLOUR INDEPENDENT SET For this reduction, we mimic the proof of Fellows et al. [11] that MULTICOLOUR CLIQUE is W[1]-complete. Let $G = (V, E)$ be the graph in an instance of p -#CLIQUE, with parameter k . Now define G' to be the cartesian product $\overline{G} \times K_k$, in which each vertex in \overline{G} (the complement of G) is “blown up” to a k -clique; the vertices of each such k -clique are given distinct colours $\{1, \dots, k\}$. It is straightforward to check that if α is the number of multicolour independent sets in G' , then the number of k -cliques in G is exactly equal to $\alpha/k!$.

p -#MULTICOLOUR INDEPENDENT SET $\leq_{\mathbf{T}}^{\text{fpt}}$ p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH The reduction follows immediately from Lemmas 2.5 and 2.6.

p -#MULTICOLOUR CONNECTED INDUCED SUBGRAPH

$\leq_{\text{T}}^{\text{fpt}}$ p -#CONNECTED INDUCED SUBGRAPH The number of multicoloured connected induced subgraphs in a graph G can be computed by inclusion-exclusion from the numbers of connected induced subgraphs in the 2^k subgraphs of G induced by different combinations of colour-classes. (Inclusion-exclusion methods have been used in a similar way in, for example, [5, 8].) Suppose the graph G is coloured with colours $[k]$, and for any $C \subseteq [k]$ let G_C be the subgraph of G induced by the vertices with colours belonging to C . Then, if $N_k(H)$ denotes the number of connected induced k -vertex subgraphs in H , the number of colourful connected induced subgraphs in G is exactly

$$\sum_{\emptyset \neq C \subseteq [k]} (-1)^{k-|C|} N_k(G_C).$$

Combining these reductions, we have p -#CLIQUE $\leq_{\text{T}}^{\text{fpt}}$ p -#CONNECTED INDUCED SUBGRAPH, and so p -#CONNECTED INDUCED SUBGRAPH is #W[1]-hard under fpt Turing reductions, as required. \square

3 Approximating p -#INDUCED SUBGRAPH WITH PROPERTY(Φ)

In contrast to the hardness result of the previous section, we now give a positive result about the approximability of a class of parameterised counting problems that includes p -#CONNECTED INDUCED SUBGRAPH, as well as, for example, the problems of counting the number of induced k -vertex Hamiltonian subgraphs, and that of counting the number of induced k -vertex non-bipartite subgraphs. This is in fact a special case of a more general result (Theorem 4.2) which will be given in Section 4 below; we prove Theorem 3.1 first as it introduces all the techniques but with slightly less complexity than is required for Theorem 4.2.

Theorem 3.1. *Let $\Phi = (\phi_1, \phi_2, \dots)$ be a monotone property, and suppose there exists a positive integer t such that, for each ϕ_k , all edge-minimal labelled k -vertex graphs (H, π) such that $\phi_k(H) = 1$ satisfy $\text{tw}(H) \leq t$. Then there is an FPTRAS for p -#INDUCED SUBGRAPH WITH PROPERTY(Φ).*

Recall from Section 1.3.2 that p -#CONNECTED INDUCED SUBGRAPH is a special case of this problem, so we obtain the following immediate corollary to Theorem 3.1.

Corollary 3.2. *There is an FPTRAS for p -#CONNECTED INDUCED SUBGRAPH.*

The proof of Theorem 3.1 is adapted from the proof of Arvind and Raman [2] that there is an FPTRAS for p -#SUB(\mathcal{H}) whenever \mathcal{H} is a class of graphs of bounded treewidth. We begin in Section 3.1 by summarising the existing results we will use, and then in Section 3.2 give a proof of the existence of an FPTRAS in the setting of Theorem 3.1.

3.1 Background

The algorithm we describe in the next section uses random sampling to count approximately, and relies heavily on the parameterised version of the Karp-Luby result [18] on this subject, given by Arvind and Raman.

Theorem 3.3 ([2, Thm. 1]). *For every positive integer n , and for every integer $0 \leq k \leq n$, let $U_{n,k}$ be a finite universe, whose elements are binary strings of length $n^{O(1)}$. Let $\mathcal{A}_{n,k} = \{A_1, \dots, A_m\} \subseteq U_{n,k}$ be a collection of $m = m_{n,k}$ given sets, with $m_{n,k} = l(k)n^{O(1)}$ for some function l , let $g : \mathbb{N} \rightarrow \mathbb{N}$ be a computable function and let $d > 0$ be a constant with the following conditions:*

1. *There is an algorithm that computes $|A_i|$ in time $g(k)n^d$, for each i , and every $\mathcal{A}_{n,k}$.*
2. *There is an algorithm that samples uniformly at random from A_i in time $g(k)n^d$, for each i , and every $\mathcal{A}_{n,k}$.*
3. *There is an algorithm that takes $x \in U_{n,k}$ as input and determines whether $x \in A_i$ in time $g(k)n^d$, for each i , and every $\mathcal{A}_{n,k}$.*

Then there is an FPTRAS for estimating the size of $A = A_1 \cup \dots \cup A_m$. In particular, for $\epsilon = 1/g(k)$, and $\delta = 1/2^{n^{O(1)}}$, the running time of the FPTRAS algorithm is $(g(k))^{O(1)}n^{O(1)}$.

In proving that there exists an FPTRAS for the problem $p\text{-\#SUB}(\mathcal{H})$ when \mathcal{H} is a class of graphs of bounded treewidth, Arvind and Raman prove two further results which we will use in Section 3.2. Firstly, they give an algorithm to compute the number of colourful copies of a k -vertex graph H (of bounded treewidth) in a k -coloured graph G ; it should be noted here that the copies of H are not necessarily induced.

Lemma 3.4 ([2, Lemma 1]). *Let $G = (V, E)$ be a graph on n vertices that is k -coloured by some colouring $f : V(G) \rightarrow [k]$, and let H be a k -vertex graph of treewidth t that is k -coloured by some colouring π such that H is colourful. Then there is an algorithm taking time $O(c^t k + n^{t+2} 2^{t^2/2})$ to exactly compute the cardinality of the set $\{K : K \text{ is a colourful } k\text{-vertex subgraph of } G \text{ and } K \text{ is colour-preserving isomorphic to } H \text{ coloured by } \pi\}$, where $c > 0$ is some constant.*

The graph H_1 with colouring ω_1 is said to be *colour-preserving isomorphic* to H_2 with colouring ω_2 if there exists an isomorphism θ from H_1 to H_2 such that, for all $u \in V(H)$, $\omega_1(u) = \omega_2(\theta(u))$. We will more generally say that a function θ from the vertices of the graph H_1 , coloured by ω_1 , to the vertices of the graph H_2 , coloured by ω_2 , is *colour-preserving* if, for all $u \in V(H_1)$, $\omega_1(u) = \omega_2(\theta(u))$.

Secondly, they describe an algorithm to sample uniformly at random from the set of colourful copies of H in G .

Lemma 3.5 ([2, Lemma 2]). *Let $G = (V, E)$ be a graph on n vertices that is k -coloured by some colouring $f : V(G) \rightarrow [k]$, and let H be a k -vertex graph of treewidth t that is k -coloured by some colouring π such that H is colourful. Then there is an algorithm taking time $O(c^{t^3}k + n^{t+O(1)}2^{t^2/2})$ time to sample uniformly at random from the set $\{K : K \text{ is a colourful } k\text{-vertex subgraph of } G \text{ under the colouring } f \text{ and } K \text{ is colour-preserving isomorphic to } H \text{ coloured by } \pi\}$.*

The approximation algorithm in [2] also uses the concept of k -perfect families of hash functions. A family \mathcal{F} of hash functions from $[n]$ to $[k]$ is said to be k -perfect if, for every subset $A \subset [n]$ of size k , there exists $f \in \mathcal{F}$ such that the restriction of f to A is injective. In the following section, we will use the following bound on the size of such a family of hash functions, proved in [1].

Theorem 3.6. *For all $n, k \in \mathbb{N}$ there is a k -perfect family $\mathcal{F}_{n,k}$ of hash functions from $[n]$ to $[k]$ of cardinality $2^{O(k)} \cdot \log n$. Furthermore, given n and k , a representation of the family $\mathcal{F}_{n,k}$ can be computed in time $2^{O(k)} \cdot n \log n$.*

3.2 An FPTRAS for p -#INDUCED SUBGRAPH WITH PROPERTY(Φ)

In this section we use Theorem 3.3 to give a proof of Theorem 3.1, that is, we show that there exists an FPTRAS for p -#INDUCED SUBGRAPH WITH PROPERTY(Φ) whenever $\Phi = (\phi_1, \phi_2, \dots)$ is a monotone property and there exists a positive integer t such that, for each ϕ_k , all edge-minimal labelled k -vertex graphs (H, π) such that $\phi_k(H) = 1$ satisfy $\text{tw}(H) \leq t$.

When considering this problem, we will take $U_{n,k}$ to be the set of all k -tuples of $[n]$; thus an element of $U_{n,k}$ can be regarded as a choice of a k -tuple of vertices in an n -vertex graph. Our goal is to approximate the cardinality of the set A , where

$$A = \{(v_1, \dots, v_k) \in V^k : \phi_k(G[v_1, \dots, v_k]) = 1\}.$$

Thus, in order to make use of Theorem 3.3 to prove the existence of an FPTRAS for p -#INDUCED SUBGRAPH WITH PROPERTY(Φ), we need to express A as a union of sets $A_1, \dots, A_{m_{n,k}}$ which satisfy the conditions of the theorem.

First, we will write A as a union of sets indexed by a family of k -perfect hash functions from V to $[k]$, which we shall regard as vertex-colourings of G . Recall from Theorem 3.6 that we can fix such a family \mathcal{F} with $|\mathcal{F}| = 2^{O(k)} \log n$. Since, by definition of a family of k -perfect hash functions, there must exist for every $U \in V^{(k)}$ some element $f_U \in \mathcal{F}$ such that the restriction of f_U to U is injective, it is clear that we can write

$$A = \bigcup_{f \in \mathcal{F}} A_f,$$

where we set

$$A_f = \{(v_1, \dots, v_k) \in V^k : \phi_k(G[v_1, \dots, v_k]) = 1 \text{ and } \{f(v_1), \dots, f(v_k)\} = [k]\}.$$

We can further write A_f as a (disjoint) union of smaller sets, conditioning on the precise injective colouring of $\{v_1, \dots, v_k\}$ under f (recall that S_k denotes the set of permutations on $[k]$):

$$A_f = \bigcup_{\sigma \in S_k} A_{f,\sigma},$$

where

$$A_{f,\sigma} = \{(v_1, \dots, v_k) \in V^k : \phi_k(G[v_1, \dots, v_k]) = 1, \text{ and, for each } 1 \leq i \leq k, f(v_i) = \sigma(i)\}.$$

In order to obtain an FPTRAS, we will need to use the assumption of Theorem 3.2, namely that Φ is monotone and that every edge-minimal labelled subgraph $(H, \pi) \in \mathcal{L}(k)$ such that $\phi_k(H, \pi) = 1$ satisfies $\text{tw}(H) \leq t$. If we write \mathcal{H}_k for the set of edge-minimal labelled subgraphs satisfying ϕ_k , this characterisation of ϕ_k implies that $\phi_k(G[v_1, \dots, v_k]) = 1$ if and only if there is some $(H, \pi) \in \mathcal{H}_k$ such that $(H, \pi) \subseteq G[v_1, \dots, v_k]$. We can therefore write $A_{f,\sigma}$ as a union over sets indexed by elements of \mathcal{H}_k :

$$A_{f,\sigma} = \bigcup_{(H,\pi) \in \mathcal{H}_k} A_{f,\sigma,(H,\pi)},$$

where

$$A_{f,\sigma,(H,\pi)} = \{(v_1, \dots, v_k) \in V^k : (H, \pi) \subseteq G[v_1, \dots, v_k] \text{ and, for each } 1 \leq i \leq k, f(v_i) = \sigma(i)\}.$$

In words, the pair of conditions in the definition above can be restated as follows: the mapping taking the vertex $\pi(i)$ of H (for each $1 \leq i \leq k$) to the vertex in $\{v_1, \dots, v_k\}$ which receives colour $\sigma(i)$ under f is, in fact, an embedding. If we then equip H with a colouring ω , where $\omega = \sigma \circ \pi^{-1}$, we can equivalently describe this embedding as the mapping which takes each vertex u of H to the unique vertex v_i such that $\omega(u) = f(v_i)$, so the mapping is the unique colour-preserving bijection from $V(H)$ to $\{v_1, \dots, v_k\}$ (with respect to colourings ω and f). With this characterisation, it is clear that the condition that $(H, \pi) \subseteq G[v_1, \dots, v_k]$ is exactly the same as the requirement that H with colouring ω is colour-preserving isomorphic to some subgraph K of $G[\{v_1, \dots, v_k\}]$. Thus,

$$A_{f,\sigma,(H,\pi)} = \{(v_1, \dots, v_k) \in V^k : \exists K \subseteq G[\{v_1, \dots, v_k\}] \text{ such that } H \text{ with colouring } \omega = \sigma \circ \pi^{-1} \text{ is colour-preserving isomorphic to } K \text{ with colouring } f\}.$$

These are the sets that will make up the collection $\mathcal{A}_{n,k}$ in Theorem 3.3; more precisely, we set

$$\mathcal{A}_{n,k} = \{A_{f,\sigma,(H,\pi)} : f \in \mathcal{F}, \sigma \in S_k, (H, \pi) \in \mathcal{H}_k\},$$

and it then follows from the reasoning above that

$$A = \bigcup \mathcal{A}_{n,k}. \quad (2)$$

Note that

$$|\mathcal{A}_{n,k}| \leq 2^{O(k)} \log n \cdot k! \cdot 2^{\binom{k}{2}} = l(k)n^{O(1)}$$

for an appropriate function l (since we can choose \mathcal{F} with $|\mathcal{F}| = 2^{O(k)} \log n$, there are $k!$ permutations on a set of size k , and there are $2^{\binom{k}{2}}$ labelled graphs on a fixed set of k vertices), as required in the premise of Theorem 3.3.

Before going on to demonstrate that this collection of sets $\mathcal{A}_{n,k}$ satisfies the three conditions of Theorem 3.3, it will be useful to make a further observations about the elements of $\mathcal{A}_{n,k}$. Note that there can be at most one subgraph $K \subseteq G[\{v_1, \dots, v_k\}]$ such that H with colouring ω is colour-preserving isomorphic to K (since the colourings determine precisely the mapping between the two graphs), so if we set

$$A'_{f,\sigma,(H,\pi)} = \{K : \begin{array}{l} K \text{ is a colourful } k\text{-vertex subgraph of } G \text{ under} \\ \text{the colouring } f, \text{ and } K \text{ is colour-preserving} \\ \text{isomorphic to } H \text{ with colouring } \omega = \sigma \circ \pi^{-1} \}, \end{array}$$

we have

$$|A_{f,\sigma,(H,\pi)}| = |A'_{f,\sigma,(H,\pi)}|. \quad (3)$$

Moreover, we can define a bijection $\theta : A'_{f,\sigma,(H,\pi)} \rightarrow A_{f,\sigma,(H,\pi)}$ by setting

$$\theta(K) = (((f|_{V(K)})^{-1} \circ \sigma)(1), \dots, ((f|_{V(K)})^{-1} \circ \sigma)(k)). \quad (4)$$

We are now ready to show that the sets $\mathcal{A}_{n,k}$ defined above do satisfy the three conditions of Theorem 3.3. The first two conditions will follow easily from results proved in [2].

Lemma 3.7. *For each $\mathcal{A}_{n,k}$ and every $A_i \in \mathcal{A}_{n,k}$, there exists an algorithm that computes $|A_i|$ in time $g_1(k)n^{d_1}$, where d_1 is an integer and $g_1 : \mathbb{N} \rightarrow \mathbb{N}$ is a computable function, for each i and every $\mathcal{A}_{n,k}$.*

Proof. Recall from (3) that, for each $A_{f,\sigma,(H,\pi)} \in \mathcal{A}_{n,k}$, we have

$$|A_{f,\sigma,(H,\pi)}| = |A'_{f,\sigma,(H,\pi)}|,$$

and so it suffices to compute $|A'_{f,\sigma,(H,\pi)}|$ in the permitted time. Since, by assumption, H has treewidth at most t , we can immediately apply Lemma 3.4 to see that there exists an algorithm to compute the cardinality of $A'_{f,\sigma,(H,\pi)}$ in time at most $O(c^{t^3}k + n^{t+2}2^{t^2/2})$ where $c > 0$ is a constant. \square

We now show that the second condition is satisfied.

Lemma 3.8. *For each $\mathcal{A}_{n,k}$ and every $A_i \in \mathcal{A}_{n,k}$, there exists an algorithm that samples uniformly at random from A_i in time $g_2(k)n^{d_2}$, where d_2 is an integer and $g_2 : \mathbb{N} \rightarrow \mathbb{N}$ is a computable function, for each i and every $\mathcal{A}_{n,k}$.*

Proof. It follows immediately from the definition of $A'_{f,\sigma,(H,\pi)}$, together with the assumption that H has treewidth at most t , that, by Lemma 3.5, there is an algorithm taking time $O(c^{t^3}k + n^{t+O(1)}2^{t^2/2})$ (where $c > 0$ is a constant) to sample uniformly at random from $A'_{f,\sigma,(H,\pi)}$. Since θ (as defined in (4)) gives a bijection from $A'_{f,\sigma,(H,\pi)}$ to $A_{f,\sigma,(H,\pi)}$, applying θ to the output of this sampling algorithm will give an element of $A_{f,\sigma,(H,\pi)}$ chosen uniformly at random; note that applying θ will require additional time depending only on k . \square

Thus, in order to apply Theorem 3.3, it remains to check that our sets satisfy the third condition; we demonstrate this in the following lemma.

Lemma 3.9. *For each $\mathcal{A}_{n,k}$ and every $A_i \in \mathcal{A}_{n,k}$, there is an algorithm that takes $\mathbf{v} \in U_{n,k}$ as input and determines whether $\mathbf{v} \in A_i$ in time $g_3(k)n^{d_3}$, where d_3 is an integer and $g_3 : \mathbb{N} \rightarrow \mathbb{N}$ is a computable function.*

Proof. For any $\mathbf{v} = (v_1, \dots, v_k) \in U_{n,k}$, in order to determine whether $\mathbf{v} \in A_i$ for any given $A_i = A_{f,\sigma,(H,\pi)}$, it suffices to check whether both the following conditions are satisfied:

1. for each $1 \leq i \leq k$, $f(v_i) = \sigma(i)$, and
2. $G[v_1, \dots, v_k] \supseteq (H, \pi)$.

The first of these two conditions can clearly be verified in time depending only on k . For the second condition we need to check, for every edge $e = uv \in E(H)$, whether $v_{\pi^{-1}(u)}x_{\pi^{-1}(v)} \in E(G)$; this can also be done in time depending only on k . The result follows immediately. \square

With these three lemmas, we can prove Theorem 3.1.

Proof of Theorem 3.1. We wish to approximate the cardinality of a set A which, by (2), can be written as a union of sets A_1, \dots, A_m (where $m_{n,k}$ is $l(k)n^{O(1)}$ for some function l). It follows from Lemmas 3.7, 3.8 and 3.9 that, if we set $g(k) = \max\{g_1(k), g_2(k), g_3(k)\}$ and $d = \max\{d_1, d_2, d_3\}$, these sets satisfy the three conditions of Theorem 3.3; it therefore follows immediately that there exists an FPTRAS for estimating the size of A , in other words there exists an FPTRAS for p -#INDUCED SUBGRAPH WITH PROPERTY(Φ). \square

4 Application to p -#GRAPH MOTIF

The GRAPH MOTIF problem was first introduced by Lacroix, Fernandes and Sagot [20] in the context of metabolic network analysis, and is defined as follows.

GRAPH MOTIF

Input: A vertex-coloured graph G and a multiset of colours M .

Question: Does G have a connected subset of vertices whose multiset of colours equals M ?

This decision problem, and a number of variations, have since been studied extensively ([3, 9, 10, 12, 15]). The problem is known to be NP-complete in general [20], and remains NP-complete even if the input is restricted so that G is a tree of maximum degree three and M is a set rather than a multiset [12]. However, the decision problem is fixed parameter tractable when parameterised by the motif size $|M|$ [12].

It is natural to consider counting versions of the GRAPH MOTIF problem, and counting the number of occurrences of a given motif in a graph has applications in determining whether a motif is over- or under-represented in a biological network with respect to the null hypothesis [23]. In [15], Guillemot and Sikora consider the following parameterised counting version of the problem.

p-#XMGM

Input: A graph $G = (V, E)$, a colouring c of V , and a multiset of colours M .

Parameter: $k = |M|$.

Question: How many k -vertex trees in G have a multiset of colours equal to M ?

The authors prove that this problem is #W[1]-hard in the case that M is a multiset, but is fixed parameter tractable when M is in fact a set (#XCGM).

In *p*-#XMGM, the output is the number of connected induced subgraphs of G having colour-set exactly equal to M , where each such subgraph is weighted by its number of spanning trees. In this section we consider a more direct translation of GRAPH MOTIF into the counting world, in which the goal is to compute simply the total number of connected induced subgraphs having the desired colour-set.

p-#GRAPH MOTIF

Input: A graph $G = (V, E)$, a colouring c of V , and a multiset of colours M .

Parameter: $k = |M|$.

Question: How many subsets $U \subset V^{(k)}$ are such that $G[U]$ is connected and the multiset of colours assigned to U is exactly M ?

We adapt results from Sections 2 and 3 to show that

- *p*-#GRAPH MOTIF is #W[1]-hard, even in the case that M is a set, and
- there exists an FPTRAS for *p*-#GRAPH MOTIF.

Our hardness result is obtained by means of a trivial reduction from *p*-#MULTICOLOUR CONNECTED INDUCED SUBGRAPH, shown to be #W[1]-complete in Section 2.1.

Theorem 4.1. *p*-#GRAPH MOTIF is #W[1]-hard, even when M is a set.

Now we show that it is possible to approximate p -#GRAPH MOTIF, for any input (G, M) . In fact, we prove that there exists an FPTRAS for the following generalisation of p -#INDUCED SUBGRAPH WITH PROPERTY(Φ) when Φ satisfies the conditions of Theorem 3.1.

p -#INDUCED COLOURED SUBGRAPH WITH PROPERTY(Φ) (p -#ICSWP(Φ))

Input: A graph $G = (V, E)$, a colouring c of V , and a multiset of colours M .

Parameter: $k = |M|$.

Question: What is the cardinality of the set $\{(v_1, \dots, v_k) \in V^k : \phi_k(G[v_1, \dots, v_k]) = 1 \text{ and } \{c(v_1), \dots, c(v_k)\} = M\}$?

Theorem 4.2. *Let $\Phi = (\phi_1, \phi_2, \dots)$ be a monotone property, and suppose there exists a positive integer t such that, for each ϕ_k , all edge-minimal labelled k -vertex graphs (H, π) such that $\phi_k(H) = 1$ satisfy $\text{tw}(H) \leq t$. Then there exists an FPTRAS for p -#ICSWP(Φ).*

Proof. Once again, we use Theorem 3.3 to demonstrate the existence of an FPTRAS for this problem. As before, we will take $U_{n,k}$ to be the set of k -element subsets of n . To make use of Theorem 3.3, we need to express

$$B = \{(v_1, \dots, v_k) \in V^k : \phi_k(G[v_1, \dots, v_k]) = 1 \text{ and } \{c(v_1), \dots, c(v_k)\} = M\}$$

as the union of some collection of sets $\mathcal{B}_{n,k}$ that satisfy the conditions of the theorem. Applying the same reasoning as in the proof of Theorem 3.1 (and using the same notation), we see that

$$B = \bigcup \{B_{f,\sigma,(H,\pi)} : f \in \mathcal{F}, \sigma \in S_k, (H, \pi) \in \mathcal{H}_k\},$$

where

$$B_{f,\sigma,(H,\pi)} = \{(v_1, \dots, v_k) \in V^k : (H, \pi) \subseteq G[v_1, \dots, v_k], \{c(v_1), \dots, c(v_k)\} = M, \text{ and, for each } 1 \leq i \leq k, f(v_i) = \sigma(i)\}.$$

Now, if we set \mathcal{D} to be the set of all bijective mappings $d : [k] \rightarrow M$, we can write

$$B_{f,\sigma,(H,\pi)} = \bigcup_{d \in \mathcal{D}} B_{f,\sigma,(H,\pi),d},$$

where

$$B_{f,\sigma,(H,\pi),d} = \{(v_1, \dots, v_k) \in V^k : (H, \pi) \subseteq G[v_1, \dots, v_k] \text{ and, for each } 1 \leq i \leq k, f(v_i) = \sigma(i) \text{ and } c(v_i) = (d \circ f)(v_i)\}.$$

So if we set

$$\mathcal{B}_{n,k} = \{B_{f,\sigma,(H,\pi),d} : f \in \mathcal{F}, \sigma \in S_k, (H, \pi) \in \mathcal{H}_{\phi_k}, d \in \mathcal{D}\},$$

we have

$$B = \bigcup \mathcal{B}_{n,k}.$$

Note that $|\mathcal{B}_{n,k}| \leq k^k \cdot |\mathcal{A}_{n,k}|$ (with $\mathcal{A}_{n,k}$ as in the proof of Theorem 3.1), and so we clearly have that $m_{n,k} = \tilde{l}(k)n^{O(1)}$ for some function \tilde{l} . It remains to check that the sets $B_j \in \mathcal{B}_{n,k}$ satisfy the conditions of Theorem 3.3.

To see that the first two conditions hold, we make use of Lemmas 3.7 and 3.8. Observe that the set $B_{f,\sigma,(H,\pi),d}$, calculated with respect to the graph G and its colouring c , is precisely equal to the set $A_{f,\sigma,(H,\pi)}$ (as defined in the proof of Theorem 3.1) if instead of considering the graph G we consider the graph

$$G_{f,d} = G[\{v \in V : c(v) = (d \circ f)(v)\}].$$

Note that, given f and d , the graph $G_{f,d}$ can be computed from G in time $O(n^2)$, so it follows from Lemmas 3.7 and 3.8 that

- for each $\mathcal{B}_{n,k}$ and every $B_j \in \mathcal{B}_{n,k}$, there is an algorithm that computes $|B_j|$ in time $g_1(k)n^{d_1+1}$, and
- for each $\mathcal{B}_{n,k}$ and every $B_j \in \mathcal{B}_{n,k}$, there is an algorithm that samples uniformly at random from B_j in time $g_2(k)n^{d_2+1}$.

Thus it remains only to check that the third condition holds. For any k -tuple $(v_1, \dots, v_k) \in V^k$ and any set $B_{f,\sigma,(H,\pi),d} \in \mathcal{B}_{n,k}$, we know by Lemma 3.9 that we can check in time $g_3(k)n^{d_3}$ whether (v_1, \dots, v_k) belongs to the related set $B_{f,\sigma,(H,\pi)}$. But $(v_1, \dots, v_k) \in B_{f,\sigma,(H,\pi),d}$ if and only if we have both that $(v_1, \dots, v_k) \in B_{f,\sigma,(H,\pi)}$ and that, for each $1 \leq i \leq k$, $c(v_i) = (d \circ f)(v_i)$; the time required to check this second condition clearly depends only on k . Hence there exists a function $g_4 : \mathbb{N} \rightarrow \mathbb{N}$ so that, for each $\mathcal{B}_{n,k}$ and every $B_j \in \mathcal{B}_{n,k}$, there is an algorithm that takes $\mathbf{v} \in U_{n,k}$ as input and determines whether $\mathbf{v} \in B_j$ in time $g_4(k)n^{d_3}$.

Hence, setting $g = \max\{g_1(k), g_2(k), g_4(k)\}$ and $d = \max\{d_1, d_2, d_3\}$, all the conditions of Theorem 3.3 are satisfied, and therefore there exists an FPTRAS for p -#ICSWP(Φ). \square

This result easily implies the existence of an FPTRAS for p -#GRAPH MOTIF.

Corollary 4.3. *There exists an FPTRAS for p -#GRAPH MOTIF.*

Proof. Setting $\Phi = \Phi^{\text{conn}}$ (as in (1)), it is clear both that Φ satisfies the conditions of Theorem 4.2 and also that the output of p -#ICSWP(Φ) will be exactly $k!$ times the number of connected induced k -vertex subgraphs whose vertices have multiset of colours equal to M (with the overcounting due to the number of distinct possible labellings of a k -vertex subgraph). Thus we know from Theorem 4.2 that there exists an FPTRAS for p -#ICSWP(Φ) in this situation, and so obtain an FPTRAS for p -#GRAPH MOTIF simply by dividing the output of the first algorithm by $k!$. \square

5 Conclusions and Open Problems

We have shown that the problem p -#CONNECTED INDUCED SUBGRAPH is #W[1]-hard, but that on the other hand there exists an FPTRAS for a more general problem p -#INDUCED SUBGRAPH WITH PROPERTY(Φ), where Φ is a monotone property such that the edge-minimal graphs satisfying Φ all have bounded treewidth. We then adapted these results to show that a natural counting version of the problem GRAPH MOTIF is #W[1]-hard, but has an FPTRAS.

We finish with two natural related open questions.

1. Are all (non-trivial) special cases of the class of problems covered by Theorem 3.1 #W[1]-hard?
2. Does there exist any problem p -#INDUCED SUBGRAPH WITH PROPERTY(Φ), where Φ is a monotone property but the edge-minimal graphs that satisfy Φ do not all have bounded treewidth, such that p -#INDUCED SUBGRAPH WITH PROPERTY(Φ) admits an FPTRAS?

References

- [1] Noga Alon, Raphael Yuster, and Uri Zwick, *Color-coding*, Journal of the ACM **42** (1995), no. 4, 844–856.
- [2] V. Arvind and Venkatesh Raman, *Approximation algorithms for some parameterized counting problems*, ISAAC 2002 (P. Bose and P. Morin, eds.), LNCS, vol. 2518, Springer-Verlag Berlin Heidelberg, 2002, pp. 453–464.
- [3] N. Betzler, M.R. Fellows, C. Komusiewicz, and R. Niedermeier, *Parameterized algorithms and hardness results for some graph motif problems*, Proceedings of the 19th Annual Symposium on Combinatorial Pattern Matching (CPM), 2008, pp. 31–43.
- [4] Yijia Chen and J. Flum, *On parameterized path and chordless path problems*, Computational Complexity, 2007. CCC '07. Twenty-Second Annual IEEE Conference on, 2007, pp. 250–263.
- [5] Yijia Chen, Marc Thurley, and Mark Weyer, *Understanding the complexity of induced subgraph isomorphisms*, Automata, Languages and Programming (Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfssdóttir, and Igor Walukiewicz, eds.), Lecture Notes in Computer Science, vol. 5125, Springer Berlin Heidelberg, 2008, pp. 587–596.
- [6] Radu Curticapean, *Counting matchings of size k is #W[1]-hard*, Automata, Languages, and Programming (Fedor V. Fomin, Rūsiņš Freivalds, Marta Kwiatkowska, and David Peleg, eds.), Lecture Notes in Computer Science, vol. 7965, Springer Berlin Heidelberg, 2013, pp. 352–363.

- [7] Radu Curticapean and Dániel Marx, *Complexity of counting subgraphs: Only the boundedness of the vertex-cover number counts*, 55th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2014, 2014.
- [8] Víctor Dalmau and Peter Jonsson, *The complexity of counting homomorphisms seen from the other side*, Theoretical Computer Science **329** (2004), no. 13, 315 – 323.
- [9] R. Dondi, G. Fertin, and S. Vialette, *Weak pattern matching in colored graphs: Minimizing the number of connected components*, Proceedings of the 10th Italian conference on Theoretical Computer Science (ICTCS), 2007, pp. 27–38.
- [10] ———, *Maximum motif problem in vertex-colored graphs*, Proceedings of the 20th Annual Symposium on Combinatorial Pattern Matching (CPM), 2009.
- [11] M. Fellows, D. Hermelin, F. Rosamond, and S. Vialette, *On the parameterized complexity of multiple-interval graph problems*, Theoretical Computer Science **410** (2009), 53–61.
- [12] Michael R. Fellows, Guillaume Fertin, Danny Hermelin, and Stéphane Vialette, *Upper and lower bounds for finding connected motifs in vertex-colored graphs*, Journal of Computer and System Sciences **77** (2011), 799.
- [13] J. Flum and M. Grohe, *The parameterized complexity of counting problems*, SIAM Journal on Computing **33** (2004), no. 4, 892–922.
- [14] ———, *Parameterized complexity theory*, Springer, 2006.
- [15] Sylvain Guillemot and Florian Sikora, *Finding and counting vertex-colored subtrees*, Algorithmica **65** (2013), no. 4, 828–844.
- [16] Pentti Haukkanen, *On meet matrices on posets*, Linear Algebra and its Applications **249** (1996), 111–123.
- [17] Mark Jerrum and Kitty Meeks, *Some hard families of parameterised counting problems*, arXiv.1310.6524 [cs.CC], October 2013.
- [18] R.M. Karp and M. Luby, *Monte-carlo algorithms for enumeration and reliability problems*, Proceedings of the 24th Annual IEEE Symposium on Foundations of Computer Science, 1983, pp. 56–64.
- [19] Subhash Khot and Venkatesh Raman, *Parameterized complexity of finding subgraphs with hereditary properties*, Theoretical Computer Science **289** (2002), no. 2, 997 – 1008.
- [20] V. Lacroix, C.G. Fernandes, and M.F. Sagot, *Motif search in graphs: Application to metabolic networks*, IEEE/ACM Trans Comput Biol Bioinform **3** (2006), no. 4, 360.

- [21] Catherine McCartin, *Parameterized counting problems*, Annals of Pure and Applied Logic **138** (2006), no. 13, 147 – 182.
- [22] Gian-Carlo Rota, *On the foundataions of combinatorial theory I. Theory of Möbius functions*, Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete **2** (1964), 340–368.
- [23] S. Schbath, V. Lacroix, and M. F. Sagot, *Assessing the exceptionality of coloured motifs in networks*, EURASIP J Bioinform Syst Biol.