Parameterized Complexity of the Smallest Degree Constraint Subgraph Problem

Omid Amini, Ignasi Sau, Saket Saurabh

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Parameterized Complexity of the Smallest Degree Constraint Subgraph Problem

Omid Amini — Ignasi Sau — Saket Saurabh

N° 6237

June 2007

Thème COM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6237--FR+ENG

Parameterized Complexity of the Smallest Degree Constraint Subgraph Problem

Omid Amini* † , Ignasi Sau * , Saket Saurabh ‡

Thème COM — Systèmes communicants
Projets MASCOTTE

Rapport de recherche n° 6237 — June 2007 — 20 pages

Abstract: In this paper we initiate the study of finding an induced subgraph of size at most k with minimum degree at least d. We call this problem Minimum Subgraph of Minimum Degree $_{\geq d}$ (MSMD$_d$). For $d = 2$, it corresponds to finding a shortest cycle of the graph. The problem is strongly related to the Dense k-Subgraph problem and is of interest in practical applications.

We show that the MSMS$_d$ is fixed parameter intractable for $d \geq 3$ in general graphs by showing it to be W[1]-hard by a reduction from Multi-Color Clique. On the algorithmic side, we show that the problem is fixed parameter tractable in graphs which excluded minors and graphs with bounded local tree-width. In particular, this implies that the problem is fixed parameter tractable in planar graphs, graphs of bounded genus and graphs with bounded maximum degree.

Key-words: Parameterized complexity, FPT algorithms, Dense subgraphs, Graph minors, Excluded minor.

This work has been partially supported by European project IST FET AEOLUS and COST 293 GRAAL.

* Projet Mascotte, CNRS/INRIA/UNSA, INRIA Sophia-Antipolis, 2004 route des Lucioles BP 93, 06902 Sophia-Antipolis Cedex, France {omid.amini,ignasi.sau}@sophia.inria.fr

† École Polytechnique, Palaiseau, France

‡ Institute of Mathematical Sciences, Chennai, INDIA saket@imsc.res.in

Complexité paramétrique du problème du sous-graphe de plus petite taille sous contraintes de degré

Résumé : Le but de cet article est d'introduire et d'étudier le problème suivant, appelé MSMD_d: Étant donné un graphe G, trouver le plus petit sous-graphe induit ayant le degré minimum au moins d, pour d un nombre entier fixé (pour $d = 2$, cela revient à trouver le cycle de plus petite taille). Ce problème est fortement lié au problème de sous-graphe le plus dense, et a des applications pratiques.

Nous démontrons que pour un graphe général, MSMS_d n'est pas FPT pour $d \geq 3$ en prouvant qu'il est dur pour la classe W[1] des problèmes paramétriques. La réduction est faite à partir du problème de la Clique Multi-colorée. Du point de vue algorithmique, on prouve que le problème est FPT pour la classe des graphes mineur-exclus et la classe des graphes de largeur arborescente localement bornée. En particulier, ceci démontre que le problème est FPT pour différentes classes de graphes: les graphes planaires, les graphes de genre borné, et les graphes de degré borné, ...

On obtient aussi des résultats analogues pour quelques variations du problème.

Mots-clés : Complexité paramétrique, Algorithme FPT, Sous-graphe dense, Mineur de graphe.

1 Introduction

In our first course on algorithms, the first few graph algorithms we learn are breadth first-search (BFS), depth first-search (DFS), and to test whether a graph is acyclic using BFS. Then we move on to more complex algorithms and learn how to find the ***smallest subgraph of degree at least two***, i.e. we are talking about a polynomial time algorithm to find a shortest cycle in the graph. The goal of this paper is to initiate an algorithmic study of the generalization of this natural problem in the realm of parameterized complexity. More precisely, the problem we study is defined as follows:

> MINIMUM SUBGRAPH OF MINIMUM DEGREE $_{\geq d}$ (MSMD$_d$): Given a graph $G = (V, E)$ and positive integers d and k, k being the parameter, the problem is to check whether there exists a subset $S \subseteq V$, with $|S| \leq k$, such that $G[S]$ has minimum degree at least d.

Besides generalizing the girth problem, our motivations for studying this problem are the followings: First, the problem is closely related to the well studied DENSE k-SUBGRAPH problem; Secondly, from practical applications, the problem is motivated by its close connection to Traffic Grooming Problem. We briefly explain these two connections:

- The DENSE k-SUBGRAPH problem: We first define the density of a graph $G = (V, E)$. The ***density*** of G, $\rho(G)$ is its edges-to-vertices ratio, that is $\rho(G) := \frac{|E|}{|V|}$. More generally, for any subset $S \subset V$, we call ***density*** of S, $\rho_G(S)$ or simply $\rho(S)$, the density of the induced graph on S, i.e. $\rho(S) := \rho(G[S])$. The problem is formulated as follows:

  > DENSE k-SUBGRAPH (DkS): Given a graph $G = (V, E)$, find a subset S of vertices, with $|S| = k$, such that $\rho(S)$ is maximized.

  First of all, note that NP-hardness of DkS easily follows from hardness of CLIQUE. On the other hand, if we do not fix the size of S, then finding a densest subgraph of G reduces to the MAX-FLOW MIN-CUT problem, and hence can be solved in polynomial time, see Chapter 4 of [14] for more details.

  Now we show how MSMD$_d$ is related to DkS. Suppose we are looking for an induced subgraph $G[S]$ of size at most k and with density at least ρ. In addition assume that S is minimal, i.e. no subset of S has density greater than $\rho(S)$. First of all, note that every vertex of $G[S]$ has degree at least $\rho/2$. To see this, observe that if there is a vertex v with degree strictly smaller than $\rho/2$, then removing v from S results in a subgraph of density greater than $\rho(S)$ and a smaller size, contradicting the minimality of S. Secondly, if we have an induced subgraph $G[S]$ of minimum degree at least ρ then it is a subset of density at least $\rho/2$. These two observations together show that, modulo a constant factor, looking for a densest subgraph of G of size at most k is equivalent to looking for the largest possible value of ρ for which $MSMD_\rho$ returns YES for the parameter k.

The Dense k-Subgraph problem has attracted a lot of attention mainly in approximation algorithms, see [9], [5], [13] and [2] for more details.

- Traffic grooming in optical networks is the process of packing several small traffic flows into larger units, which can be then processed as single entities. For example, in a network using both time-division and wavelength-division multiplexing, flows which are destined to a common node can be put on the same wavelength, allowing them to be dropped by a single optical add-drop multiplexer. The objectives of grooming are to improve bandwidth utilization and to minimize the equipment cost of the network. In WDM optical networks, the most accepted criterion is to minimize the number of electronic terminations, namely the number of SONET ADMs. See [7] for a general survey on grooming. It has been recently proved that *traffic grooming* in optical networks can be reduced (modulo polylogarithmic factors) to DkS, or equivalently to MSMD$_d$, see [1]. In fact, in graph theoretical terms, the problem can be translated to partitioning of the edges of a given request graph into subgraphs satisfying certain load constraints, and the objective is to minimize the total number of vertices of the subgraphs of the partition. Hence, in this context, partitioning a given set of edges while minimizing the total number of vertices is where, the DkS or equivalently MSMD$_d$ problem comes into plays.

Parameterized Complexity is a recent approach to deal with intractable computational problems having some parameters that can be relatively small with respect to the input size. This area has been developed extensively during the last decade. For decision problems with input size n, and a parameter k, the goal is to design an algorithm with runtime $f(k)n^{O(1)}$ where f is a function of k alone. Problems having such an algorithm are said to be fixed parameter tractable (FPT). There is also a theory of parameterized intractability using which one can identify parameterized problems that are unlikely to admit fixed parameter tractable algorithms. There is a hierarchy of intractable parameterized problem classes above FPT, the important ones are:

$$\text{FPT} \subseteq \text{M}[1] \subseteq \text{W}[1] \subseteq \text{M}[2] \subseteq \text{W}[2] \subseteq \cdots \subseteq \text{W}[\text{P}] \subseteq \text{XP}$$

The book by Downey and Fellows [6] provides a good introduction to the topic of parameterized complexity. The principal analogue of the classical intractability class NP is W[1], which is a strong analogue, because a fundamental problem complete for W[1] is the k-Step Halting Problem for Nondeterministic Turing Machines (with unlimited nondeterminism and alphabet size); this completeness result provides an analogue of Cook's Theorem in classical complexity. A convenient source of W[1]-hardness reductions is provided by the result that k-Clique is complete for W[1]. The principal "working algorithmic" way of showing that a parameterized problem is unlikely to be fixed-parameter tractable is to prove W[1]-hardness. The important property of a parameterized reduction between parameterized problems Π and Π' is that the input (x, k) to Π should be transformed to input (x', k') for Π', so that the receiving parameter k' is a function only of the parameter k for the source problem. The time taken for reduction could be $f(k)n^{O(1)}$ where f is a function of k

alone. For recent developments see the books by Flum and Grohe [11] and by Niedermeier [15].

In the first part of this paper we show that MSMD_d is not fixed parameter tractable by showing it to be W[1]-hard for any $d \geq 3$ in general graphs. In general the parameterized reductions are very stringent because of parameter preserving requirements of the reduction and requires a lot of technical care. Our reduction is based on a new methodology emerging in parameterized complexity so called *multi-color clique edge representation* and has proved useful in showing various problems to be W[1]-hard recently [3]. We first spell out a step by step procedure to use this methodology which could be used as a template for future purposes. Then we exemplify this methodology with the reduction for MSMD_d problems. Our reduction is robust in the sense that many similar problems can be shown to be W[1]-hard with just minor modifications.

After showing MSMD_d to be W[1]-hard in general graphs, we focus on graph classes where the problem can be shown to be FPT. In the second part of the paper we show that MSMD_d problems are fixed parameter tractable on all graphs excluding a fixed minor M, which in particular means that these problems are FPT on planar graphs and graphs with bounded genus. Our algorithms use dynamic programming over graphs with bounded treewidth and a few deep results concerning the clique decomposition of M-minor free graphs developed by Robertson and Seymour in their Graph Minor Theory [17]. Finally, we would like to stress that our dynamic programming over graphs with bounded (local) treewidth is very generic and can handle any variations on degree constraints subgraph problem with simple changes.

Let G be a graph. We use $V(G)$ and $E(G)$ to denote the vertices and the edges of G respectively. We simply write V and E if the graph is clear from the context. For $V' \subseteq V$ we define an *induced subgraph* $G[V'] = (V', E')$, where $E' = \{uv \in E : u, v \in V'\}$. Let $v \in V$, we denote by $N(v)$ the *neighborhood* of v, namely $N(v) = \{u \in V : uv \in E\}$. The *closed neighborhood* $N[v]$ of v is $N(v) \cup \{v\}$. In the same way we define $N[S]$ for $S \subseteq V$ as $N[S] = \cup_{v \in S} N[v]$ and $N(S) = N[S] \setminus S$. We define the *degree* of vertex v in G as the number of vertices incident to v in G. Namely, $d(v) = |\{u \in V(G) : uv \in E(G)\}|$.

2 Fixed Parameter In-Tractability of MSMD_d for $d \geq 3$

In this section we give a W[1]-hardness reduction for MSMD_d. Before, we formally define parameterized reductions:

Definition 1 *Let Π, Π' be two parameterized problems. We say that Π is (uniformly many:1) reducible to Π' if there is an algorithm Φ which transforms (x, k) into $(x', g(k))$ in time $f(k)|x|^{\alpha}$, where $f, g : \mathbb{N} \to \mathbb{N}$ are arbitrary functions and α is a constant independent of k, so that $(x, k) \in \Pi$ if and only if $(x', g(k)) \in \Pi'$. Here (x, k) and (x', k') are instances of Π and Π' respectively.*

Our reduction is from Multi-Color Clique, which is known to be W[1]-complete [10] (by a simple reduction from the ordinary Clique) and is based on the methodology known as *multi-color edge representation*. The Multi-Color Clique problem is defined as follows:

> Multi-color Clique : **I**nput: a graph G together with a proper k-coloring of the vertices of G. The problem is parameterized by k. **Q**uestion: Is there a k-clique in G consisting of exactly one vertex of each color?

Consider an instance $G = (V, E)$ of Multi-color Clique with its vertices colored with the set of colors $\{c_1, \cdots, c_k\}$. Let $V[c_i]$ denote the set of vertices of color c_i. Let $e = \{u, v\}$ be an edge of G with $u \in V[c_i]$ and $v \in V[c_j]$, and $i < j$. We first replace each such edge e by a 2-circuit, that is we replace e by two arcs $e^f = (u, v)$ and $e^b = (v, u)$. The arc $e^f = (u, v)$ (resp. $e^b = (v, u)$) will be called *forward* (resp. *backward*). By abuse of notation, we also call this digraph G. Let $E[c_i, c_j]$ be the set of arcs $e = (u, v)$, with $u \in V[c_i]$ and $v \in V[c_j]$, for $1 \leq i \neq j \leq k$. We also assume that $|V[c_i]| = N$ for all i, and that $|E[c_i, c_j]| = M$ for all $i \neq j$, i.e. we assume that the color classes of G, and also the arc sets between them, have uniform sizes. For a simple justification of this assumption, we can reduce Multi-color Clique to itself, taking a union of $k!$ disjoint copies of G, one for each permutation of the color set.

In this methodology, the basic encoding bricks correspond to arcs of G which we call *arc gadget* (in our reduction, these are graphs of regular degree $d - 1$, which we represent by $\mathcal{C}_{e^x}$ for $x \in \{f, b\}$, see Subsections 2.1 and 2.2). We generally have three kinds of gadgets that we call **Selection**, **Coherence** and **Match** Gadget. These are all engineered together to get an overall reduction gadget for the problem. In the optimal solution to the problem, the **Selection Gadget** ensures that exactly one arc gadget is selected among arc gadgets corresponding to arcs going from color classes $V[c_i]$ to another color class $V[c_j]$. Let $V[c_i]$ be a fixed color class. The **Coherence Gadget** makes sure that the out-going arcs from $V[c_i]$, corresponding to the selected arc gadgets, have a common vertex in $V[c_i]$. That is all the arcs corresponding to these selected arc gadgets emanates from the same vertex in $V[c_i]$. Finally, **Match Gadget** ensures that if we have selected an arc gadget corresponding to an arc (u, v) from $V[c_i]$ to $V[c_j]$ then the arc gadget selected from $V[c_j]$ to $V[c_i]$ corresponds to (v, u). That is both of e^f and e^b are selected together. In what follows, we show how to adhere to this general strategy and obtain a reduction from Multi-color Clique to MSMD$_d$ for $d \geq 3$. To simplify the presentation, we first describe our reduction in details for the case $d = 3$. The required modifications for the case $d \geq 4$ is then explained in Subsection 2.2.

2.1 Gadget Construction and Hardness Result for the Case $d = 3$

We have $k(k-1)$ clusters of gadgets. One cluster each for the set of arcs $E[c_i, c_j]$, for $1 \leq i \neq j \leq k$. Remember that an arc $a \in E[c_i, c_j]$ is forward if $i < j$, and is backward if $i > j$.

Forward Selection Gadget: If $i < j$: For each arc $e^f = (u, v) \in E[c_i, c_j]$ we have a cycle $\mathcal{C}_{e^f}$ of length $2 + 2(k-2) + 3$ with vertex set:

- *selection vertices:* e^f_{s1}, e^f_{s2}, and e^f_{s3};
- *coherence vertices:* e^f_{ch1r}, e^f_{ch2r}, for all $r \in \{1, \ldots, k\}$ and $r \neq i, j$; and
- *match vertices:* e^f_{m1} and e^f_{m2}.

Now we add a new vertex A_{c_i,c_j}, and connect it to all the selection vertices of the cycles $\mathcal{C}_{e^f}$ for all $e \in E[c_i, c_j]$. In other words, we add all the edges of the form $\{A_{c_i,c_j}, e^f_{s1}\}, \{A_{c_i,c_j}, e^f_{s2}\}$, and $\{A_{c_i,c_j}, e^f_{s3}\}$ where $e^f \in E[c_i, c_j]$ and e^f_{s1}, e^f_{s2} and e^f_{s3} are on $\mathcal{C}_{e^f}$. This completes the construction of the forward selection gadget between c_i and c_j, and we call it $\mathcal{S}_{i,j}$. Recall that $i < j$.

Backward Selection Gadget: If $i > j$: For each arc $e^b = (u, v) \in E[c_i, c_j]$ we have a cycle $\mathcal{C}_{e^b}$ of length $2 + 2(k-2) + 3$ with vertex set:

- *selection vertices:* e^b_{s1}, e^b_{s2}, and e^b_{s3};
- *coherence vertices:* e^b_{ch1r}, e^b_{ch2r}, for all $r \in \{1, \ldots, k\}$ and $r \neq i, j$; and
- *match vertices:* e^b_{m1} and e^b_{m2}.

Now we add a new vertex A_{c_i,c_j} and add all the edges from it to all the selection vertices of cycles $\mathcal{C}_{e^b}$ where $e^b \in E[c_i, c_j]$. In other words, as in the previous case, we add all the edges $\{A_{c_i,c_j}, e^b_{s1}\}, \{A_{c_i,c_j}, e^b_{s2}\}$, and $\{A_{c_i,c_j}, e^b_{s3}\}$ where $e \in E[c_i, c_j]$, and e^b_{s1}, e^b_{s2} and e^b_{s3} are on $\mathcal{C}_{e^b}$. This completes the construction of the backward selection gadget and we call it $\mathcal{S}_{i,j}$. Recall that $i > j$ and so there is no confusion with the last definition.

Coherence Gadget: Fix an i, $1 \leq i \leq k$. Let us consider all the selection gadgets of the form $\mathcal{S}_{i,p}$, $p \in \{1, \cdots, k\}$ and $p \neq i$. For any $u \in V[c_i]$, and any two indices $1 \leq p \neq q \leq k$, $p \neq i$, $q \neq i$ we add two new vertices u_{pq} and u_{qp}, and a new edge $\{u_{pq}, u_{qp}\}$. For every arc $e = (u, v) \in E[c_i, c_p]$, with $u \in V[c_i]$, we pick the cycle $\mathcal{C}_{e^x}$, $x \in \{f, b\}$ depending on whether e is forward or backward, and add two edges of the form $\{e_{ch1q}, u_{pq}\}$ and $\{e_{ch2q}, u_{pq}\}$. Similarly, for an arc $e = (u, w) \in E[c_i, c_q]$, with $u \in V[c_i]$, we pick the cycle $\mathcal{C}_{e^x}$, $x \in \{f, b\}$, and add two edges $\{e_{ch1p}, u_{qp}\}$ and $\{e_{ch2p}, u_{qp}\}$, see Figure 1.

Match Gadget: For the two arcs $e^f = (u, v)$ and $e^b = (v, u)$, we consider the two cycles $\mathcal{C}_{e^f}$ and $\mathcal{C}_{e^b}$ corresponding to e^f and e^b. Now, we add two new vertices e^* and e_*, a *matching edge* $\{e^*, e_*\}$, and all the edges of the form $\{e^f_{m1}, e^*\}$, $\{e^f_{m2}, e^*\}$, $\{e^b_{m1}, e_*\}$ and $\{e^b_{m2}, e_*\}$ where e^f_{m1}, e^f_{m2} are match vertices on $\mathcal{C}_{e^f}$, and e^b_{m1} , e^b_{m2} are match vertices on $\mathcal{C}_{e^b}$.

This completes the construction of all the gadgets, see Figure 1. We call this graph $\mathscr{G}_G$.

Now, we show the desired reduction through a sequence of simple claims.

Claim 1 *Let G be an instance of* MULTI-COLOR CLIQUE, *and $\mathscr{G}_G$ be the graph we constructed above. If G has a multi-colored k-clique then $\mathscr{G}_G$ has a 3-regular subgraph of size $k' = (3k+1)k(k-1)$.*

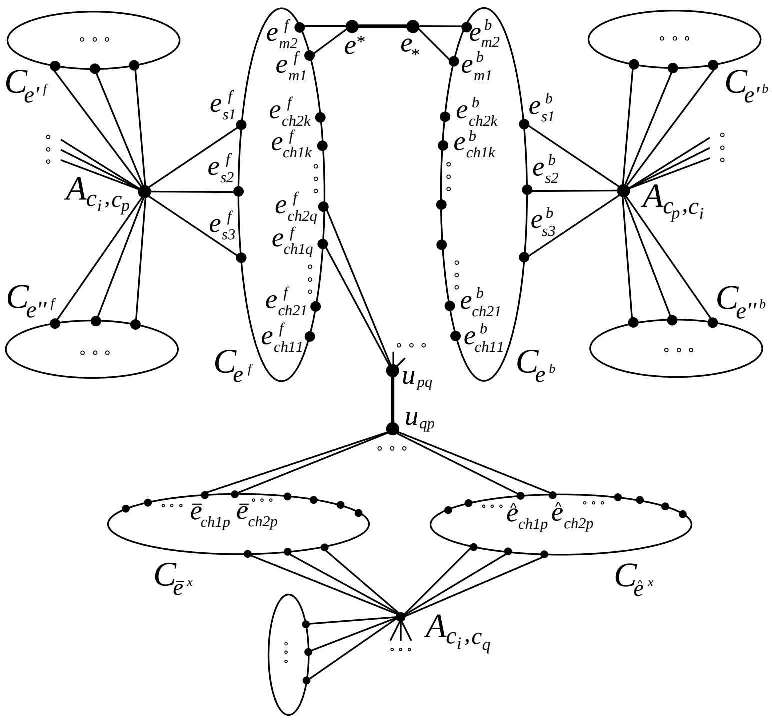

Figure 1: Gadget used in the reduction of the proof of Theorem 1

Proof: Let ω be a multi-color clique of size k in G. Now for every edge $e \in E(\omega)$, select the corresponding cycles $\mathcal{C}_{e^f}$, $\mathcal{C}_{e^b}$ in $\mathscr{G}_G$. Let us define S as follows

$$S = \bigcup_{e \in \omega, x \in \{f,b\}} N_{\mathscr{G}_G}[V(C_{e^x})]$$

where $N[A]$ is the union of A and all the neighborhood of vertices in A. It is straightforward to check that $\mathscr{G}_G[S]$ is a 3-regular subgraph of $\mathscr{G}_G$. $\square$

Claim 2 *Any subgraph of $\mathscr{G}_G$ of minimum degree at least three should contain one of the cycles $\mathcal{C}_{e^x}$, $x \in \{b, f\}$, corresponding to arc gadgets.*

Proof: First of all, if such a subgraph of $\mathscr{G}_G$ intersects a cycle $\mathcal{C}_{e^x}$ then it must contain all of its vertices. Secondly, if we remove all the vertices corresponding to edge gadgets in $\mathscr{G}_G$, then the remaining graph is a forest. Hence, these two facts together imply that any subgraph of $\mathscr{G}(G)$ of minimum degree at least three should intersect at least one cycle $\mathcal{C}_{e^x}$ corresponding to an arc gadget, and hence contains $\mathcal{C}_{e^x}$. $\square$

Claim 3 *If $\mathscr{G}_G$ contains a subgraph of size $k' = (3k+1)k(k-1)$, and of minimum degree at least three, then G has a multi-colored k-clique.*

Proof: Let $H = G[S]$ be a subgraph of size k' and with minimum degree at least three. Now, by Claim 2, S must contain all the vertices of a cycle corresponding to one arc gadget. Further more, notice that once we have a vertex of a cycle in S, then all the vertices of this cycle and their neighbors belong to S to ensure the degree condition in H. Without loss of generality, let $\mathcal{C}_{e^f}$ be this cycle, and suppose that it belongs to the gadget $\mathcal{S}_{i,j}$, i.e $e \in E[c_i, c_j]$ and $i < j$. Notice that, by construction, it forces some of the other vertices to belong to S. First, its match vertices force the cycle $\mathcal{C}_{e^b}$ of $\mathcal{S}_{j,i}$ to be in S. The coherence vertices of $\mathcal{C}_{e^f}$ force S to contain at least one cycle in $\mathcal{S}_{i,l}$, for all $l \in \{1, \cdots, k\}$, $l \neq i$. They in turn force S to contain at least one cycle in each other gadget $\mathcal{S}_{p,q}$ for all $p \neq q \in \{1, \ldots, k\}$. The selection vertices of each such cycle in $\mathcal{S}_{p,q}$ force S to contain $A_{p,q}$. But because of our condition on the size of S ($|S| = k'$), we can easily check that we can select only one cycle gadget from each of the gadgets $\mathcal{S}_{p,q}$. Let E' be the set of edges in $E(G)$ corresponding to arc cycles gadgets selected in S. First of all, because of the match vertices, once $e^f \in E'$ it forces $e^b \in E'$. On the other hand, because S is minimum, for each $i \leq k$, all the corresponding arcs in E' from $V[c_i]$ to other $V[c_p]$'s have a common vertex in $V[c_i]$. To see this, let us take two arcs $e = (u, v) \in (E[c_i, c_p] \cap E')$ and $e' = (u', w) \in (E[c_i, c_q] \cap E')$. Now the four vertices u_{pq}, u_{qp}, u'_{pq} and u'_{qp} belong to S. If u is different from u' then it would imply that S has at least 2 elements more than the expected size k', which would contradict the condition on the size of S. All these facts together imply that $G[V(E')]$ forms a multi-colored k-clique in the original graph G. $\square$

Claims 1 and 3 together yield:

Theorem 1 MSMD$_3$ *is* W[1]*-hard.*

In the next subsection, we show how to modify the proof of Theorem 1 for larger values of d.

2.2 Generalizing the W[1]-hardness of MSMD$_d$ for $d \geq 4$

Here, we generalize the reduction given in Subsection 2 for $d \geq 4$. The main idea is to change the role of the cycles $\mathcal{C}_e$ by a $(d-1)$-regular graph of appropriate size. We show below all the necessary changes in the construction of the gadgets to ensure that the same proof for $d = 3$ works for $d \geq 4$.

Let us take $\mathcal{C}$ to be a $(d-1)$-regular graph of size $(d-1) + (d-1)(k-2) + d$. As before, we replace each edge e by two arcs e^f and e^b. For each arc $e^x \in E[c_i, c_j]$, we add a copy of $\mathcal{C}$, that we call $\mathcal{C}_{e^x}$, with vertex set:

- *selection vertices:* $e^x_{s1}, e^x_{s2}, \cdots, e^x_{sd}$;
- *coherence vertices:* $e^x_{ch1r}, \cdots, e^x_{ch(d-1)r}$, for all $r \in \{1, \ldots, k\}$, $r \neq i, j$; and
- *match vertices:* $e^x_{m1}, \cdots, e^x_{m(d-1)}$.

Selection Gadgets for $d \geq 4$: W.l.g. suppose that $x = f$. As before, we add a vertex A_{c_i,c_j}, and for all arc $e^f \in E[c_i, c_j]$ we add all the edges from A_{c_i,c_j} to all the selection vertices of the graph $\mathcal{C}_{e^f}$. We again call this gadget $\mathcal{S}_{i,j}$.

Coherence Gadget for $d \geq 4$: Fix an i, $1 \leq i \leq k$. Now let consider all the selection gadgets of the form $\mathcal{S}_{i,p}$, $p \in \{1, \cdots, k\}$ and $p \neq i$.

For any $u \in V[c_i]$, and any two indices $p \neq q \leq k$, $p, q \neq i$, we add a new edge $\{u_{pq}, u_{qp}\}$. For every arc $e = (u, v) \in E[c_i, c_p]$, with $u \in V[c_i]$, we pick the graph $\mathcal{C}_{e^x}$, $x \in \{f, b\}$, depending on how e is forward or backward, and add $d - 1$ edges of the form $\{e_{ch1q}, u_{pq}\}, \{e_{ch2q}, u_{pq}\}, \ldots, \{e_{ch(d-1)q}, u_{pq}\}$. Similarly for an edge $e = (u, w) \in E[c_i, c_q]$, with $u \in V[c_i]$, we pick the graph $\mathcal{C}_{e^x}$, $x \in \{f, b\}$, and add $d - 1$ edges of the form $\{e_{ch1p}, u_{qp}\}, \ldots, \{e_{ch(d-1)p}, u_{qp}\}$.

Match Gadget for $d \geq 4$: For the two arcs $e^f = (u, v)$ and $e^b = (v, u)$, we consider the two graphs $\mathcal{C}_{e^f}$ and $\mathcal{C}_{e^b}$ corresponding to e^f and e^b. Now we add a matching edge $\{e^*, e_*\}$ and add all the edges of the form $\{e^f_{m1}, e^*\}, \ldots, \{e^f_{m(d-1)}, e^*\}$ and $\{e^b_{m1}, e_*\}, \ldots, \{e^b_{m1}, e_*\}$ where e^f_{mi}, e^b_{mi} are match vertices of $\mathcal{C}_{e^f}$ and of $\mathcal{C}_{e^b}$ respectively.

This completes the construction. It is not very hard to see that a proof similar to that of Theorem 1 could be given by showing that G, an instance of multi-color clique, has a multi-colored clique of size k if and only if $\mathscr{G}_G$ has a subgraph of size $k' = dk + 1$ with minimum degree d. Because of lack of space, we just state the following theorem.

Theorem 2 MSMD$_d$ *is* W[1]*-hard for all* $d \geq 3$.

If we replace the requirement of finding an induced subgraph of size at most k of degree at least d in MSMD$_d$ with the requirement of finding a d-regular "induced subgraph" (resp. "subgraph") H of G of size at most k, we get MI-d-RSP (resp. M-d-RSP) problem. Notice that the minimum subgraph of degree at least d in the proofs of Theorems 1 and 2 turns out to be an induced subgraph of regular degree d in $\mathcal{G}_G$. As a consequence we obtain the following corollary:

Corollary 1 MI-d-RSP *and* M-d-RSP *are* W[1]*-hard for all* $d \geq 3$.

3 FPT algorithms for Graphs with Excluded Minors

In this section we show that MSMD$_d$ is FPT for all the graphs excluding a fixed minor M. We first give necessary definitions required to handle these classes of graphs in the next subsection and then in the following subsections we give our parametrized algorithms for MSMD_d for any $d \geq 3$.

3.1 Preliminaries

A *tree-decomposition* of a graph $G = (V, E)$ is a pair $(T, \mathcal{X})$, where $T = (I, F)$ is a tree, and $\mathcal{X} = \{X_i\},\ i \in I$ is a family of subsets of $V(G)$, called *bags* and indexed by the nodes of T, such that

1. each vertex $v \in V$ appears in at least one bag, i.e. $\bigcup_{i \in I} X_i = V$;
2. for each $v \in V$ the set of nodes indexed by $\{i \mid i \in I,\ v \in X_i\}$ forms a subtree of T;
3. For each edge $e = (x, y) \in E$, there is an $i \in I$ such that $x, y \in X_i$.

The *width* of a tree-decomposition, denoted by $w((T, \mathcal{X}))$ equals $\max_{i \in I}\{|X_i| - 1\}$. The *tree-width* of G, denoted by $tw(G)$, is the minimum width of a tree-decomposition of G.

This definition of tree-width can been generalized to take into account the local properties of G and is called *local tree-width*. To define it formally we first need to define r-neighborhood of vertices of G. The *distance* $d_G(u, v)$ between two vertices u and v of G is the length of the shortest path in G from u to v. For $r \geq 1$, a *r-neighborhood* of a vertex $v \in V$ is defined as $N_G^r(v) = \{u \mid d_G(v, u) \leq r\}$. The *local tree-width* of a graph G is a function $ltw^G : \mathbb{N} \to \mathbb{N}$ which to every integer $r \in \mathbb{N}$ associates the maximum tree-width of an r-neighborhood of vertices of G, i.e.

$$ltw^G(r) = \max_{v \in V(G)}\{tw(G[N_G^r(v)])\}.$$

A graph class $\mathscr{G}$ has *bounded local tree-width*, if there exists a function $f : \mathbb{N} \to \mathbb{N}$ such that for each graph $G \in \mathscr{G}$, and for each integer $r \in \mathbb{N}$, we have $ltw^G(r) \leq f(r)$. For a given function $f : \mathbb{N} \to \mathbb{N}$, $\mathscr{G}_f$ is the class of all graphs G of local tree-width at most f, i.e. such that $ltw^G(r) \leq f(r)$ for every $r \in \mathbb{N}$. See [8] and [12] for more details.

3.2 Graphs with Bounded Local Tree-Width:

To obtain our results, we need the following lemma which gives the time complexity of finding a smallest induced subgraph of degree at least d in graphs with bounded tree-width.

Lemma 1 *Let G be a graph on n vertices with a tree-decomposition of width at most t and d be a positive integer. Then we can either find a smallest induced subgraph of degree at least d in G or answer no if no such subgraph exits in time $O((d+1)^t (t+1)^{d^2} n)$.*

Proof: Let $(T, \mathcal{X})$ be the given tree-decomposition. We suppose that T is a rooted tree, and that the decomposition is *nice*, which means:

- Any node has at most two children;

- For any node t with exactly two children t_1 and t_2, we have $X_t = X_{t_1} = X_{t_2}$;
- For any node t with exactly one child s we have $X_t \subset X_s$ and $|X_s| = |X_t| + 1$, or $X_s \subset X_t$ and $|X_t| = |X_s| + 1$.

Note that such a decomposition always exists and can be found in linear time, and in fact we can suppose that $|V(T)| = \mathcal{O}(n)$. As it is usual in algorithms based on tree decompositions, we propose a dynamic programming approach based on this decomposition, which at the end either produces a connected subgraph of G of minimum degree at least d and of size at most k, or decides that G does not have any such subgraph.

Now that the tree decomposition is rooted, we can speak of the subgraph defined by the subtree rooted at node i. More precisely, for any node i of T, let Y_i be the set of all vertices that appear either in X_i or in X_j for some descendant j of i. Denote by $G[Y_i]$ the graph induced by the nodes in Y_i.

Note that if i is a node in the tree and j_1 and j_2 are two children, then Y_{j_1} and Y_{j_2} are disjoint except for vertices in X_i, i.e. $Y_{j_1} \cap Y_{j_2} = X_i$. A $\{0,1,2,3,\ldots,d\}$-coloring of vertices in X_i is a function $c_i : X_i \to \{0,1,\ldots,d-1,d\}$. Let $supp(c) = \{v \in X_i | \ c(v) \neq 0\}$ be the *support* of c.

For any such $\{0,1,\ldots,d\}$-coloring c of vertices in X_i, let $a(i,c)$ be the minimum size of an induced subgraph $H(i,c)$ of $G[Y_i]$, which has degree $c(v)$ for every $v \in X_i$ with $c(v) \neq d$, and degree at least d on its other vertices. Note that $H(i,c) \cap X_i = supp(c)$. If such a subgraph does not exist, we define $a(i,c) = +\infty$.

We develop recursive formulas for $a(i,c)$. In the base case, i is a leaf of the tree decomposition. Hence $Y_i = X_i$. We would like to know the size of the minimum induced subgraph with prescribed degrees, but this is exactly $|supp(c)|$ if $G[supp(c)]$ satisfies the degree conditions, and is $+\infty$ if it does not.

In the recursive case, node i has at least one child. We distinguish cases, depending on the size of the bag of i and its number of children:

- Assume first that i has only one child, say j, $X_i \subset X_j$, and so $|X_j| = |X_i| + 1$ and $X_i = X_j \setminus \{v\}$ for some vertex v. Also, $Y_i = Y_j$, since X_i does not add any new vertices. Consider a coloring $c : X_i \to \{0,1,\ldots,d\}$. Consider the two colorings $c_0 : X_j \to \{0,1,\ldots,d\}$ and $c_1 : X_j \to \{0,1,\ldots,d\}$ of X_j, defined as follows: $c_0 = c_1 = c$ on X_i, and $c_0(v) = 0$, $c_1(v) = d$. Then we let $a(i,c) = \min\{a(j,c_0), a(j,c_1)\}$.
- Now assume that i has only one child, say again j, and that $|X_j| = |X_i| - 1$ and so $X_j \subset X_i$ and $X_j = X_i \setminus \{v\}$ for some vertex v. Also, $Y_j = Y_i \setminus \{v\}$. Let c be a coloring of X_i. It is clear that the only neighbours of v in $G[Y_i]$ are already in X_i.
  - If $c(v) \geq 1$, for any collection $\mathcal{A}$ of $c(v)$ edges in $G[X_i]$ connecting v to vertices $v_1,\ldots,v_{c(v)}$, with $c(v_i) \geq 1$ (note that such a collection may not exist at all), we consider the coloring $c_{\mathcal{A}}$ of X_j as follows: $c_{\mathcal{A}}(v_i) = c(v_i) - 1$ for any $1 \leq i \leq c(v)$, and $c_{\mathcal{A}}(w) = c(w)$ for any other vertex w. Then we define

$$a(i,c) = \min_{\mathcal{A}}\{a(j,c_{\mathcal{A}})\} + 1$$

  - If $c(v) = 0$, we simply define $a(i, c) = a(j, c)$.

  Note that we have at most $(t)^{d+1}$ choices for such a collection $\mathcal{A}$.

- In the last case, we can suppose that i has two children j_1 and j_2, and so $X_i = X_{j_1} = X_{j_2}$. Let c be a coloring of X_i, then $supp(c) \subset X_i$ is part of the subgraph we are looking for. For any vertex $v \in X_i$, we calculate the degree $d_{G[X_i]}(v)$. We suppose that v has degree d_1^v, d_2^v in $H \cap G[Y_{j_1}], H \cap G[Y_{j_2}]$ (H is the subgraph we are looking for). These degree sequences should be in such a way to guarantee the degree condition on v imposed by the coloring c. In other words, if $c(v) \leq d - 1$ then we should have $d_1^v + d_2^v - d_{G[X_i]} = c(v)$, and if $c(v) = d$, then $d_1^v + d_2^v - d_{G[X_i]} \geq d$. Every such sequence $\mathcal{D} = \{d_1^v, d_2^v | \ v \in X_i\}$ on vertices of X_i determines two colorings $c_1^{\mathcal{D}}$ and $c_2^{\mathcal{D}}$ of X_{j_1} and X_{j_2} respectively. For each such pair of colorings, let H_1 and H_2 be the minimum subgraphs with these degree constraints in $G[Y_{j_1}]$ and $G[Y_{j_2}]$ respectively. Then $H_1 \cup H_2$ satisfies the degree constraints imposed by c. We define

$$a(i, c) = \min_{\mathcal{D}} \{|H| \mid H = H_1 \cup H_2\}$$

  for all degree distributions as above. For every vertex we have at most d^2 possible degree choices for d_1^v and d_2^v. We have also $|X_i| \leq t + 1$. This implies that the minimum is taken over at most $(t+1)^{d^2}$ colorings.

As the size of our tree-decomposition is linear on n, we can determine all the values $a(i, c)$ for every $i \in V(T)$ and every coloring of X_i in time linear in n. Now return the minimum value of $a(i, c)$ computed for all colorings c, values in the set $\{0, d\}$ assigning at least one non-zero value. The time dependence on t follows from the size of bags and the choices we made using the colorings. $\square$

Theorem 3 MSMD$_d$ *is fixed parameter tractable on* $\mathscr{G}_f$ *for all* $d \geq 3$ *and the algorithm runs in time* $O((d+1)^{f(2k)}(f(2k)+1)^{d^2} n^2)$.

Proof: Given the input graph $G = (V, E) \in \mathscr{G}_f$, we first notice that if there exists an induced subgraph H of G of size at most k with degree at least d then H could be supposed to be connected. Secondly, if we know a vertex v of H then H is contained in $N_G^r[v]$ which has diameter at most $2k$. Now there exists the desired H if and only if there exists a $v \in V$ such that H is contained in $N_G^r[v]$. So to solve the problem we find tree-decomposition of $N_G^r[v]$ for all $v \in V$ of width $f(2k)$ in polynomial time and then run the algorithm of Lemma 1 to obtain the desired result. $\square$

The function $f(k)$ is known to be $3k$, $C_g gk$ and $b(b-1)^{k-1}$ for planar graphs, graphs of genus g and graphs of degree at most b respectively [8, 12]. Here C_g is a constant depending only on genus g of the graph. Now as an easy corollary to Theorem 3 we have the following.

Corollary 1 MSMD$_d$ *can be solved in* $O((d+1)^{6k}(6k+1)^{d^2}n^2)$, $O((d+1)^{2C_g gk}(2C_g gk+1)^{d^2}n^2)$ *and* $O((d+1)^{2b(b-1)^{k-1}}(2b(b-1)^{k-1}+1)^{d^2}n^2)$ *time in planar graphs, graphs of genus* g *and graphs of degree at most* b *respectively.*

3.3 M-minor free graphs

In this subsection we generalize the results of the last subsection to the class of M-minor free graphs. To present the results, we first introduce all the required definitions.

Definition 2 (CLIQUE-SUMS) *Let* $G_1 = (V_1, E_1)$ *and* $G_2 = (V_2, E_2)$ *be two disjoint graphs, and* $k \geq 0$ *an integer. For* $i = 1, 2$, *let* $W_i \subset V_i$, *form a clique of size* h *and let* G'_i *be the graph obtained from* G_i *by removing a set of edges (possibly empty) from the clique* $G_i[W_i]$. *Let* $F : W_1 \to W_2$ *be a bijection between* W_1 *and* W_2. *We define the* h-clique-sum *or the* h-sum *of* G_1 *and* G_2, *denoted by* $G_1 \oplus_{h,F} G_2$, *or simply* $G_1 \oplus G_2$ *if there is no confusion, as the graph obtained by taking the union of* G'_1 *and* G'_2 *by identifying* $w \in W_1$ *with* $F(w) \in W_2$, *and by removing all the multiple edges. The images of the vertices of* W_1 *and* W_2 *in* $G_i \oplus G_2$ *is called the* join *of the sum.*

We remark that $\oplus$ is not well defined; different choices of G'_i and the bijection F could give different clique-sums. A sequence of h-sums, not necessarily unique, which result in a graph G, is called a *clique-sum decomposition* of G.

Definition 3 (**h-nearly embeddable graphs**) *Let* Σ *be a surface with boundary cycles* $C_1, \ldots, C_h$. *A graph* G *is* h*-nearly embeddable in* Σ, *if* G *has a subset* X *of size at most* h, *called* apices, *such that there are (possibly empty) subgraphs* $G_0, \ldots, G_h$ *of* $G \setminus X$ *such that*

1. $G \setminus X = G_0 \cup \cdots \cup G_h$,
2. G_0 *is embeddable in* Σ, *we fix an embedding of* G_0,
3. $G_1, \ldots, G_h$ *are pairwise disjoint,*
4. *for* $1 \leq \cdots \leq h$, *let* $U_i := \{u_{i_1}, \ldots, u_{i_{m_i}}\} = V(G_0) \cap V(G_i)$, G_i *has a path decomposition* $(B_{ij}),\ 1 \leq j \leq m_i)$ *of width at most* h *such that*
   (a) *for* $1 \leq i \leq h$ *and for* $1 \leq j \leq m_i$ *we have* $u_j \in B_{ij}$
   (b) *for* $1 \leq i \leq h$, *we have* $V(G_0) \cap C_i = \{u_{i_1}, \ldots, u_{i_{m_i}}\}$ *and the points* $u_{i_1}, \ldots, u_{i_{m_i}}$ *appear on* C_i *in this order (either if we walk clockwise or anti-clockwise).*

The class of graphs h-nearly embeddable in a fixed surface Σ has linear local tree-width after removing the set of apices. Now the result of Robertson and Seymour is as follows:

Theorem 4 (Robertson and Seymour [17]) *For every graph* M *there exists an integer* h, *depending only on the size of* M, *such that every graph excluding* M *as a minor can be obtained by clique sums of order at most* h *from graphs that can be* h*-nearly embedded in a surface* Σ *in which* M *can not be embedded.*

Note that such a clique decomposition can be found in polynomial time, see [5].

Let G be a M-minor free graph, and $(T, \mathcal{B} = \{B_t\})$, a clique-decomposition of G given by Theorem 4 and obtained in polynomial time using the result in [5]. Given this rooted tree T, we define $A_t := B_t \cap B_{p(t)}$ where $p(t)$ is the unique parent of the vertex t in T, and $A_r = \emptyset$. Let $\hat{B}_t$ be the graph obtained from B_t by adding all possible edges between the vertices of A_t and also between the vertices of A_s, for each child s of t, making A_t and A_s's as cliques. In this way, G becomes a h-clique sum of the graphs $\hat{B}_t$, according to the above tree T and can also be viewed as a tree-decomposition given by $(T, \mathcal{B} = \{B_t\})$, where each $\hat{B}_t$ is h-nearly embeddable in a surface Σ in which M can not be embedded. Let X_t be the set of apices of $\hat{B}_t$. Then $|X_t| \leq h$, and $\hat{B}_t \setminus X_t$ has linear local tree-width. By G_t we denote the subgraph induced by all vertices of $B_t \cup_s B_s$, s being a descendant of t in T.

Again to simplify the presentation, here we give the proof of Theorem 5 for the case $d = 3$. Recall that we are looking for a subset of vertices S, of size at most k, which induces a graph $H = G[S]$ of minimum degree at least three.

Our algorithm consists of two level of dynamic programming. The top level of dynamic programming is over the clique decomposition and within each subproblems of this dynamic programming, we focus on the induced subgraph of the vertices in B_t. Our first level of dynamic programming computes the minimum size of a subgraph of G_t with a set of given degree constraints on the vertices of A_t. These constraints as before represent the degree of each vertex of A_t in the subgraph $H_t := G_t[S_t]$, the trace of H in G_t, where $S_t = S \cap V(G_t)$ and $H_t = H \cap G_t$. An integral part of this two level dynamic programming is a combinatorial bound on the tree-width as a function of parameter for each of B_t, after we have been able to remove apices X_t from B_t by making all possible choices in which they can interact with the desired solution. The next two lemmas are used later to obtain this combinatorial bound.

Lemma 2 *Let $H = G[S]$ a connected induced subgraph of G. The subgraph $\hat{B}_t[S \cap B_t]$ is connected.*

The proof follows from the properties of tree-decomposition and the fact that A_t and A_s's are cliques in $\hat{B}_t$, here s is child of t in T.

Lemma 3 *Let $H = G[S]$ be the minimum connected subgraph of G of minimum degree at least three. The subgraph $\hat{B}_t[S_t \cap B_t \setminus X_t]$ has at most $3h + 1$ connected components.*

Proof: Let $C_1, \ldots, C_r$ be the connected components of $\hat{B}_t[S_t \cap B_t \setminus X_t]$. We prove that it is impossible to have $r > 3h + 1$. Suppose the converse. For each vertex v of $X_t \cap S_t$, with degree d_v in H_t, let b_v be the minimum of d_v and 3. Choose at most b connected components, covering at least b neighbors of v in H_t. We also add the connected component which contains all the vertices of $A_t \setminus X_t$ (A_t forms a clique in $\hat{B}_t$). Let A be the union of all the vertices of these chosen connected components. For each child s of t with $A_s \cap A \neq \emptyset$, we infer that $A_s \setminus X_t \subset A$, since A_s forms a clique in $\hat{B}_t$. Let us take the union of all these vertices in S_s, $(X_t \cup A) \cap S_t$, and $S \setminus S_t$. Let H' be the induced subgraph on these vertices in G. We claim that H' has minimum degree at least three. This will give a contradiction. Indeed, assuming $r > 3h + 1$, we have $|H'| < |H|$, because there are some vertices of H_t

which are in some connected component C_i which do not intersect H'. To prove the claim, first note that the degree of each vertex v of $X_t \cap V(H')$ in H' is at least three. This is because v has degree at least b_v in H'_t. If $d_v < 3$, then v should be in A_t (if not, v has degree $d_v < 3$ in H, which is impossible), and hence v is connected to at least $3 - d_v$ vertices in $S \setminus S_t$. But $S \setminus S_t$ is included in H', and so every vertex of $X_t \cap H'$ have degree at least three in H'. Now, we prove that every other vertex of H' has also degree at least three. This is trivially true for the vertices of $H \setminus H_t$, because of $A_t \cap H \subset H'$. Evey vertex in A has the same degree in both H' and H. The reason for this is that A is the union of some connected components and no vertex of a A is to connected to any other vertex in any other component. This proves that every vertex in A has degree at least 3 in H'. To conclude, we prove that the rest of the vertices of H', i.e. the vertices of $H' \setminus (A \cup (H \setminus H_t))$, also have degree at least three in H'. Remember that all these vertices are in some S_s, for s, such that A_s has a non empty intersection with A. We claim that all these vertices have the same degree in both H and H'. To prove this, note that $H' \cap A_s = H \cap A_s$ for all such s. Indeed, $A_s \setminus X_t \subset A$, and so $A_s \subset A \cup X_t$. Let u be such a vertex. We can suppose that $u \notin X_t$. If $u \in A_s$ then clearly $u \in A$, and so we are done. If $u \in S_s \setminus A$, then every neighbor of u is in H_s. But $H_s \subset H'$ and so we are done. □

We define a *coloring* of A_t to be a function $c : A_t \cap S \rightarrow \{0, 1, 2, 3\}$. For $i < 3$, $c(v) = i$ means that the vertex v has degree i in the subgraph H_t of G_t that we are looking for and $c(v) = 3$ means that v has degree at least three in H_t. By $a(t, c)$ we denote the minimum size of a subgraph of G_t with the prescribed degrees according to c in A_t. Now we describe the different steps of our algorithm.

Recursively, starting from the leaves of T to the root, for each node $t \in V(T)$ and **for every coloring** c **of** A_t, we compute $a(t, c)$ based on the values of $a(s, c)$ where s is a child of t or store $a(t, c) = +\infty$ if no such subgraph exits. We describe the various steps involved in computing $a(t, c)$ for a fixed coloring c.

(i) We guess a subset $R_t \subseteq X_t \setminus A_t$ such that $R_t \subseteq S_t$. We have at most 2^h choices for R_t.

(ii) For each vertex v in R_t, we guess whether v is adjacent to a vertex of $B_t \setminus (R_t \cup A_t)$, i.e. we test all the two colorings $\gamma : R_t \rightarrow \{0, 1\}$ such that $\gamma(v) = 1$ means v is adjacent to a vertex of $B_t \setminus (R_t \cup A_t)$. The number of such colorings is at most 2^h. For a fixed coloring γ, we guess one vertex in $B_t \setminus (R_t \cup A_t)$ which we supppose to be in S_t, for each of the vertex v in R_t, such that $\gamma(v) = 1$. For each coloring γ, we have at most n^h choices for the new vertices which could be included in S_t. If a vertex has $\gamma(v) = 0$, it is not allowed to be adjacent to any vertex of B_t beside the vertices in $A_t \cup R_t$. Let D_t^γ be these newly chosen vertices at this level.

(iii) Now we remove all the vertices of X_t from B_t. Lemma 3 ensures that the induced graph $\hat{B}_t[S_t \cap B_t \setminus X_t]$ has at most $3h + 1$ connected components. Now we guess these connected components of $\hat{B}_t[S_t \cap B_t \setminus X_t]$ by guessing a vertex from these connected components in $B_t \setminus X_t$. Since we need to choose at most $3h+1$ vertices this way, we have at most $(3h+1)n^{3h+1}$ new choices. Let these newly chosen vertices be F_t^γ and

$$R_t^\gamma = R_t \cup D_t^\gamma \cup F_t^\gamma \cup \{v \in A_t \setminus X_t \mid c(v) \neq 0\}.$$

Let G_t^* be the graph induced by the k-neighborhood of all vertices of R_t^γ in $\hat{B}_t \setminus X_t$, i.e. $G_t^* = (\hat{B}_t \setminus X_t)[N^k(R_t^\gamma)]$.

(iv) Each connected component of G_t^* has diameter at most $2k$ in $\hat{B}_t \setminus X_t$. As $\hat{B}_t \setminus X_t$ has bounded local tree-width, this implies that G_t^* has tree-width bounded by a function of k. By the result of Demaine and Hajiaghayi [4], this function in the previous statement can be chosen to be linear.

(v) In this step we first find a tree-decomposition $(\mathcal{T}_\gamma, \{U_p\})$ of G_t^*. Now since $A_s \cap G_t^*$ is a clique it appears in a bag of this tree-decomposition. Let the node representing this bag in this tree be p. Now we make a new bag containing the vertices of $A_s \cap G_t^*$ and make it a leaf of the tree $\mathcal{T}_\gamma$ by adding a node and connecting this node to p. By abuse of notation, by s we denote this distinguished leaf containing the bag $A_s \cap G_t^*$. We also add all the vertices of A_t to all the bags of this tree-decomposition which only increases the bag size by at most h. Now we apply a dynamic programming algorithm similar to the one we used for the bounded local tree-width case. Remember that for each child s of t, we have a leaf in this decomposition with the bag $A_s \cap G_s^*$. The aim is to find an induced subgraph which respects all the choices we made at earlier steps above, and which has the minimum size.

We start from the leaves of this decomposition $\mathcal{T}_\gamma$ and go to its root. At this point we have all the values for $a(s, c')$, for all possible colorings c' of A_s, where s is a child of t (because of first level of dynamic programming). Now to compute $a(t, c)$, we apply the dynamic programming algorithm of Lemma 1 with the restriction that for each ***distinguished*** leaf s of this decomposition, we have already all the values $a(s, c)$ for all colorings of $A_s \cap G_s^*$ (we extend this coloring to all A_s by giving the zero values to vertices of $A_s \setminus G_s^*$). Note that the only difference between this dynamic programming and the one of the Lemma 1 is the way we initialize the leaves of the tree.

(vi) Return the minimum size of a subgraph with the degree constraint c on A_t, among all the subgraphs we found this way. Let $a(t, c)$ be this minimum.

(vii) If for some vertex t and a colouring $c : A_t \to \{0, 3\}$, we have $1 \leq a(t, c) \leq k$, return YES. If not, we conclude that such a subgraph does not exist.

This completes the description of the algorithm. Now we discuss the time complexity of the algorithm. Let C_M be the constant depending only on linear local tree-width of the surfaces in which M can not be embedded. For each fixed coloring c, we need $4^{C_M k}(C_M k+1)^9 n^{4h+1}$ time to obtain $a(t, c)$ where $t \in T$. Now, since the number of coloring of each A_t is at most 4^h, and size of the clique-decomposition is $O(n)$, we get the following theorem:

Theorem 5 *Let $\mathcal{C}$ be a class of graphs with excluded minor M. Then one can find an induced subgraph of size at most k with degree at least 3 in $\mathcal{C}$ in time $\mathcal{O}(4^{\mathcal{O}(k+h)}(\mathcal{O}(k))^9 n^{\mathcal{O}(1)})$ where the constants in the exponents depends only on M. That is* MSMD$_3$ *is* FPT *on the class of graphs excluding a minor M.*

Theorem 5 can be generalized to larger values of d with slight modification. We state the following theorem without a proof.

Theorem 6 *Let $\mathcal{C}$ be a class of graphs with an excluded minor M. Then one can find an induced subgraph of size at most k with degree at least 3 in $\mathcal{C}$ in time $\mathcal{O}((d+1)^{\mathcal{O}(k+h)}(\mathcal{O}(k))^{d^2} n^{\mathcal{O}(1)})$ where the constants in the exponents depends only on M. That is* MSMD$_d$ *is* FPT *on the class of graphs excluding a minor M for all $d \geq 3$.*

4 Conclusion

In this paper we introduced, MSMD$_d$ problem, a generalization of finding shortest cycle in a graph and studied it from the view point of parameterized complexity. We showed that the MSMD$_d$ is W[1]-hard in general undirected graphs for $d \geq 3$ and gave fixed parameter tractable algorithms when the input graph was either of bounded local tree-width or excluded a fixed minor M. We believe that our algorithmic initiations will trigger further research on the problem. This will help us in understanding not only this problem but also the closely related problem of DkS. The parameterized tractability of traffic grooming problem still remains open.

Acknowledgement. The authors would like to thank M. Fellows, S. Pérennes and D. Lokshtanov for insightfull discussions.

References

[1] O. Amini, S. Pérennes and I. Sau. *Hardness and Approximation of Traffic Grooming,* Rapport de Recherche INRIA, 2007, submitted.

[2] R. Andersen, *Finding large and small dense subgraphs*, http://arXiv:cs/0702032v1, 2007, submitted.

[3] B. Chor, M. Fellows, M. A. Ragan, I. Razgon, F. Rosamond and S. Snir. Connected coloring completion for general graphs: algorithms and complexity. *Proceedings* COCOON 2007, Springer-Verlag, *Lecture Notes in Computer Science*, 2007, to appear.

[4] E. Demaine, M.T. Haijaghayi, *Equivalence of Local Treewidth and Linear Local Treewidth and its Algorithmic Applications*, 2003

[5] E. Demaine, M. T. Hajiaghayi and K. C. Kawarabayashi, *Algorithmic Graph Minor Theory: Decomposition, Approximation and Coloring*, In Proceeding of the 46th Annual IEEE Symposium on Fondations of Computer Science (FOCS), Pages 637-646, October 2005.

[6] R. G. Downey and M. R. Fellows. *Parameterized Complexity.* Springer-Verlag, 1999.

[7] R. Dutta and N. Rouskas, *Traffic grooming in WDM networks: Past and future*, IEEE Network, **16**, N. 6, Pages 46-56, 2002.

[8] D. Eppstein, *Diameter and Tree-width in Minor-closed Graph Families*, Algorithmica, **27**(3-4), 275–291, 2000.

[9] U. Feige and G. Kortsarz and D. Peleg, *The Dense k-Subgraph Problem*, Algorithmica, **29**, n. 3, 410–421, 2001.

[10] M. Fellows, D. Hermelin and F. Rosamond. On the fixed-parameter intractability and tractability of multiple-interval graph properties. Manuscript, 2007.

[11] J. Flum and M. Grohe, *Parameterized Complexity Theory*, Springer-Verlag, 2006.

[12] M. Grohe, *Local Tree-width, Excluded Minors and Approximation Algorithms*, Combinatorica, **23**, n.4, 613–632, 2003.

[13] S. Khot, *Ruling out* PTAS *for graph min-bisection, densest subgraph and bipartite clique*, FOCS, 136–145, 2004

[14] E.L Lawler, *Combinatorial Optimization: Networks and Matroids,* Holt, Rinehart and Winston, 1976.

[15] R. Niedermeier, *Invitation to Fixed Parameter Algorithms.* Oxford University Press, 2006.

[16] N. Robertson and P. Seymour, *Graph minors V, Excluding a planar graph*, J. Comb. Theory, Series B, **41**, n. 2, 92–114, 1986.

[17] N. Robertson and P. Seymour, *Graph minors XVI, Excluding a non-planar graph,* J. Comb. Theory, Series B, **77** (1999) 1–27.

Contents

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399