Some results on more flexible versions of Graph Motif

Romeo Rizzi\textsuperscript{1} and Florian Sikora\textsuperscript{2,3}

\textsuperscript{1} DIMI, Università di Udine, Italy. romeo.rizzi@uniud.it
\textsuperscript{2} Université Paris-Est, LIGM - UMR CNRS 8049, France
\textsuperscript{3} Lehrstuhl für Bioinformatik, Friedrich-Schiller-Universität Jena, Germany. florian.sikora@uni-jena.de

\textbf{Abstract.} The problems studied in this paper originate from Graph Motif, a problem introduced in 2006 in the context of biological networks. Informally speaking, it consists in deciding if a multiset of colors occurs in a connected subgraph of a vertex-colored graph. Due to the high rate of noise in the biological data, more flexible definitions of the problem have been outlined. We present in this paper two inapproximability results for two different optimization variants of Graph Motif. We also study another definition of the problem, when the connectivity constraint is replaced by modularity. While the problem stays \textit{NP}-complete, it allows algorithms in \textit{FPT} for biologically relevant parameterizations.

\section{Introduction}

A recent field in bioinformatics focuses in biological networks, which represent interactions between different elements (e.g. between amino acids, between molecules or between organisms) [1]. Such a network can be modeled by a vertex-colored graph, where nodes represent elements, edges represent interactions between them and colors give functional informations on the graph nodes. Using biological networks allows a better characterization of species, by determining small recurring subnetworks, often called \textit{motifs}. Such motifs can correspond to a set of nodes realizing a same function, which may have been evolutionary preserved [22]. It is thus crucial to determine these motifs to identify common elements between species and transfer the biological knowledge.

Historically, motifs were defined by a set of nodes labels with a given topology (e.g. a path, a tree, a graph). The algorithmic problem was thus to find an occurrence of the motif in the network which respect both the label set and the given topology. This leads to problems roughly equivalent to subgraph isomorphism, a computationally difficult problem. However, in metabolic networks, similar topology can represent very different functions [17]. Moreover, in protein-protein interactions (PPI) networks, informations about the topology of motifs is often missing [5]. There is also a high rate of false positive and false negative in such networks [10]. Therefore, in some situations, topology is irrelevant, which leads to search for \textit{functional} motifs instead of \textit{topological} ones. In this
setting, we still ask for the conservation of the node labels, but we replace topology conservation by the weaker requirement that the subnetwork should form a connected subgraph of the target graph. This approach was proposed by Lacroix et al., defining Exact Graph Motif [17].

- **Input:** A graph $G = (V, E)$, a set of colors $C$, a function $col : V \rightarrow C$, a multiset $M$ over $C$, an integer $k$.
- **Output:** A subset $V' \subseteq V$ such that (i) $|V'| = k$, (ii) $G[V']$ is connected, and (iii) $col(V') = M$.

In the following, the motif is said colorful if $M$ is a set (it is a multiset otherwise). Note that this problem also has application in the context of mass spectrometry [4], and may be used in social or technical networks [3,23].

Not surprisingly, the problem remains NP-complete, even under strong restrictions (when $G$ is a bipartite graph with maximum degree 4 and $M$ is built over two colors only [11], or when $M$ is colorful and $G$ is a rooted tree of depth 2 [2] or a tree of maximum degree 3 [11]). However, for general trees and multiset motifs, the problem can be solved in $O(n^{2c+2})$ time, where $c$ is the number of distinct colors in $M$, while being W[1]-hard for the parameter $c$ [11]. We also point out that the problem can be solved in polynomial time if the number of colors in $M$ is bounded and if $G$ is of bounded treewidth [11]. It is also polynomial if $G$ is a caterpillar [2], or if the motif is colorful and $G$ is a tree where the colors appears at most two times. This last result is mentioned in [9] and can be retrieved by an easy transformation to a 2-SAT instance (chapter 4 of [23]).

The difficulty of this problem is counterbalanced by its fixed-parameter tractability when the parameter is $k$, the size of the solution [17,11,3,5,14,13] (for information about parameterized complexity, one can for example see [18]). The currently fastest algorithms in FPT for Exact Graph Motif run in $O^*(2^k)$ time for the colorful case, $O^*(4^k)$ time for the multiset case, and in both cases use polynomial space [13] (the $O^*$ notation suppresses polynomial factors). A recent paper shows that the problem is unlikely to admit polynomial kernels, even on restricted classes of trees [2].

To deal with the high rate of noise in biological data, different variants of Exact Graph Motif have been introduced. The approach of Dondi et al. requires a solution with a minimum number of connected components [8], while the one of Betzler et al. asks for a 2-connected solution [3]. As for traditional bioinformatics problems, some colors can be inserted in a solution, or conversely, some colors of the motif can be deleted in a solution [5,8,13]. Recently, Dondi et al. introduced a variant when the number of substitutions between colors of the motif and colors in the solution must be minimum [9].

Following this direction, we consider in Section 2 an approximation issue when one wants to maximize the size of the solution. In Section 3, we propose an inapproximability result when one wants to minimize the number of substitutions. Finally, we present in Section 4 a new requirement concerning the connectedness of the solution with one hardness result and two fixed-parameter tractable (FPT) algorithms. Due to space constraints, most proofs are in missing.
2 Maximizing the solution size

To deal with the high rate of noise in the biological data, one approach allows some colors of the motif to be deleted from the solution, leading to MAX GRAPH MOTIF, a problem introduced by Dondi et al. [8].

- **Input:** A graph \( G = (V, E) \), a set of colors \( C \), a function \( \text{col} : V \rightarrow C \), a multiset \( M \) over \( C \).
- **Output:** A subset \( V' \subseteq V \) such that (i) \( G[V'] \) is connected, and (ii) \( \text{col}(V') \subseteq M \).
- **Measure:** The size of \( V' \).

In a natural decision form, we are also given an integer \( k \) in the input and one looks for a solution of size \( k \) (the number of deletions is thus equal to \(|M| - k\)). The problem is known to be in the FPT class for parameter \( k \) [8, 5, 13].

Concerning its approximation, MAX GRAPH MOTIF is APX-hard, even when \( G \) is a tree of maximum degree 3, the motif is colorful and each color appears at most two times in \( G \) (in the same conditions, recall that the EXACT GRAPH MOTIF is polynomial [9]). Moreover, there is no constant approximation ratio unless \( P = NP \), even when \( G \) is a tree and \( M \) is colorful [8].

In the following, we answer an open question of Dondi et al. [8] concerning the approximation issue of the problem when \( G \) is a tree where each color occurs at most twice. To do so, we use a reduction from MAX INDEPENDENT SET, a problem stated as follows: Given a graph \( G_I = (V_I, E_I) \), find the maximum subset \( V'_I \subseteq V_I \) where there is no two nodes \( u, v \in V'_I \) such that \( \{u, v\} \in E_I \). Our proof proceeds in four steps. We first describe the construction of the instance \( \mathcal{I}' = (G, C, \text{col}) \) for MAX GRAPH MOTIF from the instance \( \mathcal{I} = (G_I) \) of MAX INDEPENDENT SET (we consider the motif as \( M = C \)). We next prove that we can construct in polynomial time a solution for \( \mathcal{I}' \) from a solution for \( \mathcal{I} \) and, conversely, that we can construct in polynomial time a solution for \( \mathcal{I} \) from a solution for \( \mathcal{I}' \). Finally, we show that if there is an approximation algorithm with ratio \( r \) for MAX GRAPH MOTIF, then there is an approximation algorithm with ratio \( r \) for MAX INDEPENDENT SET.

Before stating the reduction, consider a total order over the edges of \( G \). We then define a function \( \text{adj} : V_I \rightarrow 2^{E_I} \), giving for a node \( v \in V_I \), the ordered list of edges where \( v \) is involved (thus of size \( d(v) \), the degree of \( v \)). With this order, consider that \( \text{adj}(v)[i] \) give the \( i \)-th edge where \( v \) is involved. From the graph \( G_I = (V_I, E_I) \), we build the graph \( G = (V, E) \) as follows (see also Figure 1):

- \( V = \{r\} \cup \{v^i_i : 1 \leq i \leq |V_I|, e \in \text{adj}(v_i)\} \cup \{v^i_j : 1 \leq i \leq |V_I|, 1 \leq j \leq \lfloor |V_I|^2 \rfloor\}, \)
- \( E = \{(r, v^i_{\text{adj}(v_i)[i]}), 1 \leq i \leq |V_I|\} \cup \{\{v^i_{\text{adj}(v_i)[i]}, v^i_{\text{adj}(v_i)[i]+1}\}, 1 \leq i \leq |V_I|, 1 \leq j < d(v_i)\} \cup \{\{v^i_{\text{adj}(v_i)[d(v_i)]}, v^i_{1}\}, 1 \leq i \leq |V_I|\} \cup \{\{v^i_j, v^{i+1}_j\}, 1 \leq i \leq |V_I|, 1 \leq j < \lfloor |V_I|^2 \rfloor\}. \)

Informally speaking, \( r \) is the root of \( G \). There are \(|V_I| \) paths connected to \( r \). Each path represents a node of \( G_I \) and is of length \( d(v_i) + \lfloor |V_I|^2 \rfloor \). Observe that
For each 1 \leq i \leq |V_I|, we add \( v_i \) in \( V_I' \) iff all the nodes \( v_i', e \in \text{adj}(v_i) \) are in \( V' \). In other words, we add \( v_i \) in \( V_I' \) if the whole path corresponding to this node is in \( V' \).

These two lemmas lead to the main result of this section.
**Proposition 1.** Unless $P = NP$, there is no approximation ratio lower than $|V|^{\frac{1}{4}-\epsilon}$ for MIN GRAPH MOTIF, for any $\epsilon > 0$, even when the motif is colorful and $G$ is a tree where each color of $C'$ appears at most two times.

**Proof.** Suppose there is such a ratio $r$ for MIN GRAPH MOTIF. Then, there is an approximate solution $V'_{APX}$ which, compared to the optimal solution $V'_{OPT}$, is of size $|V'_{APX}| \geq \frac{|V'_{OPT}|}{r}$.

With Lemma 1, $|V'_{OPT}| \geq |V'_{OPT}|^2 - |V|$. We supposed $|V'_{APX}| \geq |V|$. Therefore, $|V'_{APX}| \geq |V'|^2 - |V|$. With Lemma 2, $|V'_{APX}| \geq \left(\frac{|V|^{2(r)} - 2|E|}{|V|^r}\right) - 1 = |V|^2 - 1$. Which leads to, $|V'_{APX}| \geq \left(\frac{|V|^{2(r)} - 2|E|}{|V|^r}\right) - 1 = |V|^2 - 1$.

Thereby, if there is an approximation algorithm with ratio $r$ for MIN GRAPH MOTIF, there is an approximation algorithm with ratio $r$ for MIN INDEPENDENT SET. We conclude the proof by observing that $|V| = O(|V|^3)$ and that unless $P = NP$, there is no ratio lower than $|V|^{1-\epsilon}$ for MIN INDEPENDENT SET, $\forall \epsilon > 0$ [24].

## 3 Minimizing the number of substitutions

In this section, we focus on MIN SUBSTITUTE GRAPH MOTIF, a problem recently introduced by Dondi et al. [9]. In this variant, some colors of the motif can be deleted, but the size of the solution must be equal to $|M|$. Therefore, the deleted colors must be substituted by the same number of colors.

- **Input:** A graph $G = (V, E)$, a set of colors $C$, a function $col : V \rightarrow C$, a multiset $M$ over $C$.
- **Output:** A subset $V' \subseteq V$ such that (i) $|V'| = |M|$ and (ii) $G[V']$ is connected.
- **Measure:** The number of substitutions to get $M$ from $col(V')$.

Dondi et al. [9] prove that MIN SUBSTITUTE GRAPH MOTIF is NP-hard, even when $G$ is a tree of maximum degree 4 where each color occurs at most twice and the motif is colorful. On the positive side, they prove that the problem is in the FPT class when the parameter is the size of the solution.

Unfortunately, even in restrictive conditions (when $G$ is a tree of depth 2 and the motif is colorful), we prove that there is no approximation ratio within $c \log |V|$, for a constant $c$. To prove such inapproximability result, we do an $L$-reduction from MIN SET COVER [20], a problem stated as follows: Given a set $X = \{x_1, x_2, \ldots, x_{|X|}\}$ and a collection $S = \{S_1, S_2, \ldots, S_{|S|}\}$ of subsets of $X$, find the minimum subset $S' \subseteq S$ such that every element of $X$ belongs to at...
least one member of $S'$. We denote by $e(i, j)$ the index $l$ such that $x_l$ correspond to the $j$-th element of $S_i$. We first describe the polynomial construction of $I' = (G, C, col, M)$, instance of Min Substitute Graph Motif, from $I = (X, S)$, any instance of Min Set Cover. From an instance $I$, let build $G = (V, E)$ as follows (see also Figure 2):

- $V = \{r\} \cup \{v_i : 1 \leq i \leq |S|\} \cup \{v_{i,j,t} : 1 \leq i \leq |S|, 1 \leq j \leq |S_i|, 1 \leq t \leq |S| + 1\}$,
- $E = \{(r, v_i) : 1 \leq i \leq |S|\} \cup \{\{v_i, v_{i,j,t}\} : 1 \leq i \leq |S|, 1 \leq j \leq |S_i|, 1 \leq t \leq |S| + 1\}$.

Fig. 2: Illustration of the construction of an instance of Min Substitute Graph Motif from an instance of Min Set Cover such that $X = \{x_1, x_2, x_3\}$ and $S = \{\{x_1, x_2\}, \{x_2, x_3\}, \{x_3\}\}$. For ease, only the color of each node of the graph (and not the label) is given. The associated motif is $M = \{c_r\} \cup \{c_{k,t} : 1 \leq k \leq 3, 1 \leq t \leq 4\}$. A possible solution (with two substitutions) is given in bold.

Informally speaking, $r$ is the root of a tree with $|S|$ children, corresponding to each subset of $S$. Each child $v_i, 1 \leq i \leq |S|$, got $(|S| + 1)|S_i|$ children, corresponding to $|S| + 1$ copies of each element of $S_i$. The set of colors is $C = \{c_r\} \cup \{c_i : 1 \leq i \leq |S|\} \cup \{c_{k,t} : 1 \leq k \leq |X|, 1 \leq t \leq |S| + 1\}$. The coloring function is such that the root has a unique color, i.e. $col(r) = c_r$. Each node $v_i$ is colored with the unique color corresponding to the subset of $S$, $col(v_i) = c_i, \forall 1 \leq i \leq |S|$. Each node $v_{i,j,t}$ get the color of the copy of the represented element, i.e. $col(v_{i,j,t}) = c_{e(i,j)}$. Finally, the motif is $M = \{c_r\} \cup \{c_{k,t} : 1 \leq k \leq |X|, 1 \leq t \leq |S| + 1\}$. Observe that the colors $\{c_i : 1 \leq i \leq |S|\}$ are not in the motif (which is colorful by construction). Let now show how to build a solution for $I'$ from a solution for $I$, and vice-versa.
Lemma 3. If there is a solution $S'$ for an instance $I$ of Min Set Cover, there is a solution for the instance $I'$ of Min Substitute Graph Motif with $|S'|$ substitutions.

Lemma 4. From a solution for the instance $I'$ for Min Substitute Graph Motif with at most $s$ substitutions, there is a solution for the instance $I$ for Min Set Cover of size at most $s$.

Proof. Let $V' \subseteq V$ be a solution for $I'$ such that we can obtain $M$ from $col(V')$ with at most $s$ substitutions. We can suppose that $s < |S| + 1$, otherwise, $S' = S$ is a solution of correct size. Solution for $I$ is built as follows: $S' = \{ S_i : v_i \in V' \}$. If for some $1 \leq k \leq |X|$ there is no color of the set $\{ c_{k,t} : 1 \leq t \leq |S| + 1 \}$ in the solution, it means that these $|S| + 1$ colors have all been substituted, which is a contradiction with the supposed maximum number of $s$ substitutions. Therefore, for each $1 \leq k \leq |X|$, there is at least one color from the set $\{ c_{k,t} : 1 \leq t \leq |S| + 1 \}$ in the solution. Thus, since the solution is connected, all elements of $X$ are covered by $S'$. Finally, the size of $S'$ is bounded by $s$. Indeed, since their colors are not in the motif, there are at most $s$ nodes $v_i$ in $V'$.

We can now state the main result of this section.

Proposition 2. Unless P = NP, there is no polynomial approximation algorithm for Min Substitute Graph Motif with a ratio lower than $c \log |V|$, where $c$ is a constant, even when the motif is colorful and $G$ is a tree of depth 2.

As a corollary of Proposition 2, we remark that the reduction is also an parameterized reduction. Since Min Set Cover is W[2]-hard if parameterized by the number of subsets in the solution [18], Min Substitute Graph Motif is also W[2]-hard if parameterized by the number of substitutions.

Corollary 1. Min Substitute Graph Motif is W[2]-hard when parameterized by the number of substitutions.

4 Using modularity

In this section, we introduce a variant of Exact Graph Motif, where the connectivity constraint is replaced by modularity. After a quick recall on the modules properties, we justify this new variant. The problem stays NP-hard, however, the tools offered by the modularity allow efficient algorithms.

4.1 Definitions and properties

In an undirected graph $G = (V, E)$, a node $x$ separates two nodes $u$ and $v$ iff $\{x, u\} \in E$ and $\{x, v\} \notin E$. A module $\mathcal{M}$ of a graph $G$ is a set of nodes not separated by any node of $V \setminus \mathcal{M}$. In other words, a module $\mathcal{M}$ is such that $\forall x \notin \mathcal{M}, \forall u, v \in \mathcal{M}, \{x, u\} \in E \iff \{x, v\} \in E$ [6]. The whole set of nodes $V$ and any singleton set $\{u\}$, where $u \in V$, are the trivial modules. Before stating
the definition of specific modules, let say that two modules $A$ and $B$ overlap if (i) $A \cap B \neq \emptyset$, (ii) $A \setminus B \neq \emptyset$, and (iii) $B \setminus A \neq \emptyset$. According to [6], if two modules $A$ and $B$ overlap, then $A \cap B$, $A \cup B$ and $(A \cup B) \setminus (A \cap B)$ are also modules. This allows the definition of strong modules. A module is strong if no other module overlaps it, otherwise it is weak. Therefore, two strong modules are either included into the other, either of empty intersection. A module $M \subseteq S$ is said maximal for a given set of nodes $S$ (by default the set of nodes $V$) if there is no module $M'$ s.t. $M \subset M' \subset S$. In other words, the only module which contains the maximal module $M$ is $S$.

There are three types of modules: (i) parallel, when the subgraph induced by the nodes of the module is not connected (it is a parallel composition of its connected components), (ii) series, when the complement of the subgraph induced by the nodes of the module is not connected (it is a series composition of the connected components of its complement), or (iii) prime, when both the subgraph induced by the nodes of the module and its complement are connected.

The inclusion order of the maximal strong modules defines the modular tree decomposition $T(G)$ of $G$, which is enough to store the whole set of strong modules. The tree $T(G)$ can be recursively built by a top-down approach, where the algorithm recurs on the graph induced by the considered strong module. The root of this tree is the set of all nodes $V$ while the leaves are the singleton sets $\{u\}$, $\forall u \in V$. Each node of $T(G)$ got a label representing the type of the strong module, parallel, series or prime. Children of an internal node $M$ are the maximal submodules of $M$ (i.e. they are disjoints). The modular tree decomposition can be obtained with a linear time algorithm, (e.g. the one described in [15]). We can now introduce an essential property of $T(G)$:

**Theorem 1.** ([6]) A module of $G$ is either a node of $T(G)$, either a union of children (of depth 1) of a series or parallel node in $T(G)$.

One can see strong modules as generators of the modules of $G$: the set of all modules of $G$ can be obtained from the tree $T(G)$. A crucial point to note is that there is potentially an exponential number of modules in a graph (e.g., the clique $K_n$ has $2^n$ modules), but the size of $T(G)$ is $O(n)$ (more precisely, $T(G)$ has less than $2^n$ nodes since there are $n$ leaves and no node with exactly one child). Therefore, the exponential-sized family of modules of $G$ can be represented by the linear sized tree $T(G)$.

### 4.2 When modules join Graph Motif

In the following, we investigate the algorithmic issues of other topology-free definition, when replacing the connectedness demand by modularity. Following definition of Exact Graph Motif, we introduce Module Graph Motif.

**Input:** A graph $G = (V, E)$, a set of colors $C$, a function $col : V \rightarrow C$, a multiset $M$ on $C$ of size $k$.

**Output:** A subset $V' \subseteq V$ such that (i) $V'$ is a module of $G$ and (ii) $col(V') = M$. 

This definition links the modularity demand with the motif research. The module definition implies that all the nodes in this module have a uniform relation with the set of all the other nodes outside of the module. The module nodes are indistinguishable from the outside, they are acting similarly with the other nodes of the graph.

Authors of [1] define a biological module as a set of elements having a separable function from the rest of the graph. Similarly, authors of [19] describe a biological module as a set of some elements with an identifiable task, separable from the functions of the other biological modules. Moreover, it is shown in [7] that genes with a similar neighborhood have chances to be in a same biological process. It is thus possible that set of nodes in an algorithmic module of a graph representing a biological network have a common biological function. Also note that authors of [21] describe modules in gene regulatory networks as groups of genes which obey to the same regulations, and consequently, as groups which members cannot be distinguished from the rest of the network.

Moreover, apart of using modules in a slightly different goal (in order to predict more cleverly results of PPI), Gagneur et al. [12] note that modules of a graph can join biological modules, and consider modular decomposition as a general tool for biological network analysis under different representations (oriented graphs, hyper-graphs...).

However, there is no clear definition of what is (or should be) a biological module in a network [1]. We thus claim that the approach using modular decomposition is complementary to the previous definitions of biological modules (e.g. connected occurrences or compact occurrences).

4.3 Difficulty of the problem

Unfortunately, Module Graph Motif is NP-hard, even under strong restrictions, i.e. when $G$ is a collection of paths of size three, and when the motif is colorful. Observe that under the same conditions, Exact Graph Motif is trivially polynomial-time solvable.

**Proposition 3.** Module Graph Motif is NP-Complete even if $G$ is a collection of paths of size 3 and $M$ is colorful.

**Proof (Sketch).** To prove the hardness of Module Graph Motif, we propose a reduction from the NP-complete problem Exact Cover by 3-Sets (X3C) stated as follows: Given a set $X = \{x_1, \ldots, x_{3q}\}$ and a collection $S = \{S_1, \ldots, S_{|S|}\}$ of 3-elements subsets of $X$, does $S$ contains a subcollection $S' \subseteq S$ such that each element of $X$ occurs in exactly one element of $S'$?

Let us now describe the construction of an instance $T' = (G, C, col)$ of Module Graph Motif from an arbitrary instance $T = (X, S)$ of X3C. The graph $G = (V, E)$ is built as follows: $V = \{v^i_j : 1 \leq i \leq |S|, x_j \in S_i\}$, $E = \{v^i_1, v^i_2\} \cup \{v^i_2, v^i_3\} : 1 \leq i \leq |S|$. Informally speaking, $G$ is a collection of $|S|$ paths with three nodes (recall that for each $1 \leq i \leq |S|$, $|S_i| = 3$). The set of colors is $C = \{c_i : 1 \leq i \leq |X|\}$. The coloration of $G$ is such that
col(v_i^j) = c_j. In other words, each node get the color of the represented element of X. We also consider the colorful motif as M = C.

\[ \Box \]

4.4 Algorithms for the decision problem

Even if the problem is hard under strong restrictions, the modular decomposition tree is a useful structure to design efficient algorithms. More precisely, we show in the sequel that Module Graph Motif is in the FPT class when the parameter is the size of the solution, with a better complexity than for Exact Graph Motif when the motif is a multiset. As a corollary, we show that the problem can be solved in polynomial time if the number of colors is bounded. Moreover, Module Graph Motif is still in the FPT class if a set of colors is associated to each node of the graph.

Let us first observe that asking for a strong module instead of any module in the definition of Module Graph Motif leads to a linear algorithm. Indeed, one can just browse \( T(G) \) and test if the set of colors for each strong module is equal to the motif.

Let us now show an algorithm with a time complexity of \( O^*(2^k) \), where \( k \) is the size of the solution, for Module Graph Motif, even if the motif is a multiset. To the best of our knowledge, we do not know an algorithm with a time complexity lower than \( O^*(4^k) \) for Exact Graph Motif when the motif is a multiset.

**Proposition 4.** There is an algorithm for Module Graph Motif with a time complexity of \( O(2^k|V|^2) \) and a space complexity of \( O(2^k|V|) \), where \( k \) is the size of the motif and of the solution.

**Proof (Sketch).** Since \( |M| = k \), observe that there are at most \( 2^k \) different multisets \( M' \) such that \( M' \subseteq M \). We first build in polynomial time the modular tree decomposition \( T(G) \) from \( G \). We repeat the following algorithm for each node \( M \) of \( T(G) \).

We start by testing if the set of the colors of \( M \) is exactly equal to the motif \( M \). If it is the case, the algorithm terminates. Otherwise, if \( M \) is a series or parallel node, a module can be a union of its children. Given an arbitrary order on its \( t \) children, denote by \( \text{Child}(M)[i] \) the \( i \)-th child of \( M \). We then delete all children \( M' \) of \( M \) such that \( \text{col}(M') \not\subseteq M \), where \( \text{col}(M') \) is the set of colors of the nodes of \( M' \). Indeed, such a child cannot be in a solution considering \( M \). We note that the set of colors for each child correspond to a multiset \( M' \subseteq M \). Since any union of children of \( M \) is a module of \( G \), it is thus a potential solution. We propose to test by dynamic programming if such union corresponds to a solution for \( M \). We build a table \( D(i, M') \), for \( 0 \leq i \leq t \) and \( M' \subseteq M \). Therefore, \( D \) has \( t + 1 \) lines and \( 2^k \) columns. We fill this table as follows:

\[
D(0, M') = \text{True} \text{ if } M' = \{0, \ldots, 0\}, \text{ False otherwise,} \\
D(i, M') = D(i - 1, M') \lor D(i - 1, M' \setminus \text{col(Child}(M)[i])) \text{ if } i \leq t, M' \subseteq M.
\]
The algorithm returns True iff $D(t, M) = True$. Informally speaking, the first part of the computation of $D(i, M')$ ignores the i-th child of $M$ while the second part add this child into the potential solution.

Corollary 2. **Module Graph Motif** is in FPT when parameterized by $(k, |C|)$.

Proof. Note that, by definition of the motif $M$, for each color $c \in C$, $occ_M(c) \leq k$. Thus, the number of multisets $M'$ such that $M' \subseteq M$ is less than $k^{|C|}$. The time complexity of the algorithm in Proposition 4 is bounded by $O(k^{|C|}|V|^2)$.

This corollary is quite surprising and shows a fundamental difference with **Exact Graph Motif**. Indeed, recall that this problem is NP-complete, even when the motif is built over two different colors [11]. Let us now show that even when a set of colors is associated to each node of the graph, the problem is still in the FPT class. It is indeed biologically relevant to consider many functions for a same reaction in a metabolic network or to consider more than one homology for a protein in a PPI network [17,3].

A version of **Exact Graph Motif** with a set of colors for each graph node as been defined, and thus, we can introduce the analogous problem **List-Colored Module Graph Motif**.

- **Input**: A graph $G = (V, E)$, an integer $k$, a set of colors $C$, a multiset $M$ over $C$, a function $col : V \rightarrow 2^C$ giving a set of colors for each node of $V$.
- **Output**: A subset $V' \subseteq V$ such that (i) $|V'| = k$, (ii) $V'$ is a module of $G$ and (iii) there is a bijection $f : V' \rightarrow M$ such that $\forall v \in V', f(v) \in col(v)$.

Proposition 5. **List-Colored Module Graph Motif** is in the FPT class.

4.5 Open problems

Clearly, the noise in the biological data implies that searching exact occurrences of modules is too restrictive to consider a practical evaluation. Indeed, only one false positive or one false negative can suppress a potential solution. Adding flexibility as in variants for **Exact Graph Motif** seems essential. Deletions can be easily handled, but what about the insertions of colors or of nodes not in a module? It would also be interesting to know if **Module Graph Motif** is W[1]-hard if the parameter is the number of colors in the motif as for **Exact Graph Motif**, or if using modularity change the complexity class.

The complexity of the algorithm of Proposition 5 is not satisfying for practical issues. We believe that this complexity can be improved by the use of multilinear monomials detection [16].

References

1. E. Alm and A. P. Arkin. Biological Networks. *Curr Opin Struct Biol*, 13(2):193–202, 2003.
2. A. M. Ambalath et al. On the Kernelization Complexity of Colorful Motifs. In IPEC, volume 6478 of LNCS, pages 14–25. Springer, 2010.
3. N. Betzler et al. Parameterized Algorithms and Hardness Results for Some Graph Motif Problems. In CPM, volume 5029 of LNCS, pages 31–43. Springer, 2008.
4. S. Böcker, F. Rasche, and T. Steijger. Annotating Fragmentation Patterns. In WABI, volume 5724 of LNCS, pages 13–24. Springer, 2009.
5. S. Bruckner, F. Hüffner, R. M. Karp, R. Shamir, and R. Sharan. Topology-Free Querying of Protein Interaction Networks. J Comput Bio, 17(3):237–252, 2010.
6. M. Chein, M. Habib, and M.-C. Maurer. Partitive hypergraphs. Discrete math, 37(1):35–50, 1981.
7. M. Costanzo et al. The Genetic Landscape of a Cell. Science, 327(5964):425–431, 2010.
8. R. Dondi, G. Fertin, and S. Vialette. Complexity issues in vertex-colored graph pattern matching. J Discr Algo, 9(1):82–99, 2011.
9. R. Dondi, G. Fertin, and S. Vialette. Finding Approximate and Constrained Motifs in Graphs. In CPM, volume 6661 of LNCS. Springer, 2011. To appear.
10. A. M. Edwards et al. Bridging structural biology and genomics: assessing protein interaction data with known complexes. Trends Gen, 18(10):529 – 536, 2002.
11. M. R. Fellows et al. Sharp tractability borderlines for finding connected motifs in vertex-colored graphs. In ICALP, volume 4596 of LNCS, pages 340–351, 2007.
12. J. Gagneur, R. Krause, T. Bouwmeester, and G. Casari. Modular decomposition of protein-protein interaction networks. Genome Biol, 5(8):R57, 2004.
13. S. Guillemot and F. Sikora. Finding and counting vertex-colored subtrees. Algorithmica. 10.1007/s00453-011-9600-8.
14. S. Guillemot and F. Sikora. Finding and counting vertex-colored subtrees. In MFCS, volume 6281 of LNCS, pages 405–416. Springer, 2010.
15. M. Habib, F. d. Montgolfier, and C. Paul. A Simple Linear-Time Modular Decomposition Algorithm for Graphs, Using Order Extension. In SWAT, volume 3111 of LNCS, pages 187–198. Springer, 2004.
16. I. Koutis and R. Williams. Limits and Applications of Group Algebras for Parameterized Problems. In ICALP, volume 5555 of LNCS, pages 653–664, 2009.
17. V. Lacroix, C. G. Fernandes, and M.-F. Sagot. Motif search in graphs: application to metabolic networks. IEEE/ACM T Comput Bi, 3(4):360–368, 2006.
18. R. Niedermeier. Invitation to Fixed Parameter Algorithms. Lecture Series in Mathematics and Its Applications. Oxford University Press, 2006.
19. E. Ravasz et al. Hierarchical Organization of Modularity in Metabolic Networks. Science, 297(5586):1551–1555, 2002.
20. R. Raz and S. Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In STOC, pages 475–484. ACM, 1997.
21. E. Segal et al. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. Nat Genet, 34(2):166–176, 2003.
22. R. Sharan and T. Ideker. Modeling cellular machinery through biological network comparison. Nat biotechnol, 24(4):427–433, April 2006.
23. F. Sikora. Aspects algorithmiques de la comparaison d’éléments biologiques. PhD thesis, Université Paris-Est, 2011. (in French).
24. D. Zuckerman. Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number. Theor Comput, 3(1):103–128, 2007.
Appendix

Lemma 1. If there is a solution \( V'_I \subseteq V_I \) for \( I \), then there is a solution \( V' \subseteq V \) for \( I' \) such that \(|V'| \geq |V'_I| \cdot |V_I|^2\).

Proof. We build \( V' \) as follows: \( V' = \{ r \} \cup \{ v^e_i, v^j_i : v_i \in V'_I, e \in \text{adj}(v_i), 1 \leq j \leq |V_I|^2 \} \). In other words, we add in \( V' \) the root \( r \) of the tree and all the paths corresponding to the nodes of \( V'_I \) (see also Figure 1).

Let us prove that \( V' \) is a solution for \( I' \) such that \(|V'| \geq |V'_I| \cdot |V_I|^2\). Since the root is in the solution, \( G[V'] \) is connected. Moreover, colors of \( V' \) are all distinct, therefore the solution is colorful. Indeed, if there are \( u, v \in V' \) such that \( \text{col}(u) = \text{col}(v) \), then \( \{u, v\} \in E \), which is a contradiction since \( V'_I \) is a solution for Max Independent Set. Finally, we bound the size of \( V' \) by observing that for each \( v \in V'_I \), we add the set of nodes in the path corresponding to \( v \), which is of size \( d(v) + |V_I|^2 \geq |V_I|^2 \).

Lemma 2. If there is a solution \( V' \subseteq V \) for \( I' \), then there is a solution \( V'_I \subseteq V_I \) for \( I \) such that \(|V'_I| \geq \left\lceil \frac{|V'| - 2|E_I| - 1}{|V_I|^2} \right\rceil \).

Proof. For each \( 1 \leq i \leq |V_I| \), we add \( v_i \) in \( V'_I \) iff all the nodes \( v^e_i, 1 \leq j \leq |V_I|^2 \) and \( v^e_i, e \in \text{adj}(v_i) \) are in \( V' \). In other words, we add \( v_i \) in \( V'_I \) if the whole path corresponding to this node is in \( V' \).

Let us prove that \( V'_I \) is a solution for \( I \) such that \(|V'_I| \geq \left\lceil \frac{|V'| - 2|E_I| - 1}{|V_I|^2} \right\rceil \). If there are \( v_i, v_j \in V'_I \) such that \( \{v_i, v_j\} = e \in E \), then \( v^e_i \) and \( v^e_j \) are in \( V' \). It is impossible since \( \text{col}(v^e_i) = \text{col}(v^e_j) \) and since all the colors of \( V' \) must be distinct to be a solution for Max Graph Motif. Consequently, \( V'_I \) is an independent set. There are \( \left\lceil \frac{|V'| - 2|E_I| - 1}{|V_I|^2} \right\rceil \) whole paths in \( V' \). Indeed, by removing \( 2|E_G| + 1 \) to the whole number of nodes in the solution, we bound the number of nodes of type \( v^e_i \) (recall that \( |V| = 1 + 2|E_I| + |V_I|^2 \)).

Lemma 3. If there is a solution \( S' \) for an instance \( I \) of Min Set Cover, there is a solution for the instance \( I' \) of Min Substitute Graph Motif with \(|S'| \) substitutions.

Proof. Let \( S' \subseteq S \) be a solution for \( I \). Given a total order on the subsets of \( S \), for each \( 1 \leq k \leq |X| \), denote by \( S^k_{\text{min}} \) the subset such that (i) \( S^k_{\text{min}} \subseteq S' \), and (ii) \( S^k_{\text{min}} \) is the first subset of \( S' \) where \( x_k \) is. Moreover, for each \( S_i \), denote by \( f_i \) the smallest index \( j \) of \( v_{i,j,t} \) such that \( S_i = S^{(t,j)}_{\text{min}} \).

The solution \( V' \) is built as follows: \( V' = \{ r \} \cup \{ v_i : S_i \in S' \} \cup \{ v_{i,j,t} : S_i = S^{(t,j)}_{\text{min}}, j = f_i, 2 \leq t \leq |S_i| + 1 \} \cup \{ v_{i,j,t} : S_i = S^{(t,j)}_{\text{min}}, j \neq f_i, 1 \leq t \leq |S_i| + 1 \} \). Less formally, we put in the solution the root, the set of nodes representing subsets \( S_i \) of \( S' \), also with the \(|S| + 1 \) copies of each node representing an \( x_k \) (the one in the subset with minimal index in the solution), except for the element \( x_k \) of \( X \) with the lower index in \( S^k_{\text{min}} \), where only \(|S| \) copies are in the solution.
The graph $G[V']$ is connected since the nodes $v_{i,j,t}$ are in the solution if and only if the node $v_i$ is also in the solution. Moreover, a node $v_i$ is in the solution if there is a $k$ such that $S_i = S_{k_{\text{min}}}^k$. There is thus an integer $f_i$ for which only $|S|$ copies of $v_{i,f_i,t}$ are in the solution. Therefore, by construction, the color $c_{e(i,f_i),t}$, which is in the motif, is substituted in the solution by $c_i$. On the whole, there are $|S'|$ substitutions, since the other colors of the motif are in the solution.

**Proposition 2.** Unless $P = NP$, there is no polynomial approximation algorithm for **Min Substitute Graph Motif** with a ratio lower than $c \log |V|$, where $c$ is a constant, even when the motif is colorful and $G$ is a tree of depth 2.

**Proof.** The proof comes directly from the Lemmas 3 and 4, and because there is no approximation algorithm with a ratio lower than $c \log |X|$ for **Min Set Cover**, unless $P = NP$ [20]. Observe that the parameter is strictly the same between the two instances $I$ and $I'$, therefore, it is an L-reduction.

**Proposition 3.** **Module Graph Motif** is NP-Complete even if $G$ is a collection of paths of size 3 and $M$ is colorful.

**Proof.** **Module Graph Motif** is in NP since given a set $V' \subseteq V$, one can check in polynomial-time if $V'$ is a module and if the colors of $C$ appears exactly once if $V'$. To prove its hardness, we propose a reduction from **Exact Cover by 3-Sets** (X3C). This special case of **Set Cover** is known to be NP-complete. Recall that X3C is stated as follows: Given a set $X = \{x_1, x_2, \ldots, x_{3q}\}$ and a collection $S = \{S_1, \ldots, S_{|S|}\}$ of 3-elements subsets of $X$, does $S$ contains a subcollection $S' \subseteq S$ such that each element of $X$ occurs in exactly one element of $S'$. Size of $X$ must be a multiple of three since a solution is a set of triplets where each element of $X$ must appears exactly once.

Let us now describe the construction of an instance $I' = (G, C, \text{col})$ of **Module Graph Motif** from an arbitrary instance $I = (X, S)$ of X3C (see also Figure 3). The graph $G = (V, E)$ is built as follows: $V = \{v_i^1 : 1 \leq i \leq |S|, x_j \in S_i\}$, $E = \{|v_i^1, v_i^2| : 1 \leq i \leq |S|\}$. Informally speaking, $G$ is a collection of $|S|$ paths with three nodes (recall that for each $1 \leq i \leq |S|, |S_i| = 3$).

![Fig. 3: The graph $G$ built from $X = \{x_1, x_2, \ldots, x_6\}$ (thus with $q = 2$) and $S = \{\{x_1, x_3, x_5\}, \{x_1, x_2, x_4\}, \{x_2, x_4, x_6\}, \{x_2, x_3, x_6\}\}$ (only the colors of the node are written). By construction, the set of colors asked in any solution is $C = \{c_1, c_2, \ldots, c_6\}$.](image-url)
The set of colors is $C = \{c_i : 1 \leq i \leq |X|\}$. The coloration of $G$ is such that $\text{col}(v_j^i) = c_j$. In other words, each node get the color of the represented element of $X$. We also consider the colorful motif as $M = C$.

Let us now prove that if there is a solution for an instance $I$ of $X3C$, then there is solution for the instance $I'$ of $\text{Module Graph Motif}$. Given a solution $S' \subseteq S$ for $I$, a solution $V'$ for $I'$ is built as follows: $V' = \{v_j^i : S_i \in S', x_j \in S_i\}$. Informally speaking, the solution contains the set of paths corresponding to the chosen triplets in the solution for $X3C$. The set $V'$ is a module, and by definition of a solution for $I$, each color of $\{c_i : 1 \leq i \leq |X|\}$ appears exactly once in $V'$.

Conversely, let us now prove that there is a solution for the instance $I$ of $X3C$ if there is a solution for the instance $I'$ of $\text{Module Graph Motif}$. First observe that since $\prod q \geq 1$, then $|X| \geq 3$ and therefore $C \geq 3$. A module of size greater or equal than three in a collection of paths of size three must be a union of paths of size three. Indeed, suppose by contradiction that there is a module $M$ of size greater than three which is not a union of paths of size three. There is thus a node $u \in M$ such that at least one of its neighbor $v \in N(u)$ is not in $M$ and $v$ separates $u$ from another node of $M$. Therefore, $M$ is not a module. The solution is built as follows: $S' = \{S_i : v_j^i \in V'\}$. Since the solution $V'$ is a union of paths of size three, each triplet $S_i$ is either completely chosen in the solution $S'$, either absent. Moreover, since $V'$ is a solution, colors of $V'$ appears exactly once. Therefore, each element of $X$ appears exactly once in $S'$.

**Proposition 4.** There is an algorithm for $\text{Module Graph Motif}$ with a time complexity of $O(2^k|V|^2)$ and a space complexity of $O(2^k|V|)$, where $k$ is the size of the motif and of the solution.

**Proof.** Since $|M| = k$, observe that there are at most $2^k$ different multisets $M'$ such that $M' \subseteq M$. We first build in polynomial time the modular tree decomposition $T(G)$ from $G$. We repeat the following algorithm for each node $M$ of $T(G)$.

---

Fig. 4: A sample graph in a) and the corresponding modular tree decomposition in b). Nodes of the tree are either series ($s$), parallel ($p$), prime ($\text{prime}$) or leaves.
We start by testing if the set of the colors of $\mathcal{M}$ is exactly equal to the motif $M$. If it is the case, the algorithm terminates. Otherwise, if $\mathcal{M}$ is a series or parallel node, a module can be a union of its children. Given an arbitrary order on its $t$ children, denote by $\text{Child}(\mathcal{M})[i]$ the $i$-th child of $\mathcal{M}$. We then delete all children $\mathcal{M}'$ of $\mathcal{M}$ such that $\text{col}(\mathcal{M}') \not\subseteq M$, where $\text{col}(\mathcal{M}')$ is the set of colors of the nodes of $\mathcal{M}'$. Indeed, such a child cannot be in a solution considering $\mathcal{M}$. We note that the set of colors for each child correspond to a multiset $\subseteq M$.

Since any union of children of $\mathcal{M}$ is a module of $G$, it is thus a potential solution. We propose to test by dynamic programming if such union corresponds to a solution for $M$. We build a table $D(i, M')$, for $0 \leq i \leq t$ and $M' \subseteq M$. Therefore, $D$ has $t + 1$ lines and $2^k$ columns. We fill this table as follows:

\[
D(0, M') = \text{True} \text{ if } M' = \{0, \ldots, 0\}, \text{ False otherwise,}
\]

\[
D(i, M') = D(i - 1, M') \lor D(i - 1, M' \setminus \text{col}(\text{Child}(\mathcal{M})[i])) \text{ if } i \leq t, M' \subseteq M.
\]

The algorithm returns $\text{True}$ iff $D(t, M) = \text{True}$. Informally speaking, the first part of the computation of $D(i, M')$ ignores the $i$-th child of $\mathcal{M}$ while the second part add this child into the potential solution.

The time and space complexities of the dynamic programming are $O(2^k|V|)$ since $D$ is of size at most $2^k|V|$ and the computation time for each element is constant. Therefore, since the dynamic programming is launched in the worst case on each node of $\mathcal{T}(G)$, the whole time complexity is $O(2^k|V|^2)$.

It remains to show the correctness of the dynamic programming. Suppose the existence of a module $\mathcal{M}'$ such that $\text{col}(\mathcal{M}') = M$. Then, either $\mathcal{M}'$ is a strong module represented in a node of $\mathcal{T}(G)$, or it is a union of $j$ modules $\mathcal{M}'_1, \mathcal{M}'_2, \ldots, \mathcal{M}'_j$, children of a module $\mathcal{M}$. Therefore, $M \setminus \{\text{col}(\mathcal{M}'_1) \cup \text{col}(\mathcal{M}'_2) \cup \cdots \cup \text{col}(\mathcal{M}'_j)\} = \{0, 0, \ldots, 0\}$, then $D(t, M) = \text{True}$.

Conversely, if there is a module $\mathcal{M}$ such that $D(t, M) = \text{True}$, then there is a union of the children of $\mathcal{M}$ such that the set of colors of these children is equal to $M$.

\[\square\]

**Proposition 5.** List-Colored Module Graph Motif is in the FPT class.

**Proof.** We first build the modular tree decomposition $\mathcal{T}(G)$ from $G$. We repeat the following algorithm for each node $\mathcal{M}$ of $\mathcal{T}(G)$.

If $\mathcal{M}$ has less than $k$ nodes, we look for a bijection between the colors of $\mathcal{M}$ and $M$. To do so, we try all the possible combinations. In the worst case, there are $c^k$ such combinations, where $c$ is the number of different colors in $M$ (thus $c \leq k$).

In the following, we thus can consider $\mathcal{M}$ with more than $k$ nodes. If it is a prime node, we can ignore it since this node cannot be a solution for a motif of size $k$. Otherwise, it is a series or parallel node, and a union of the children can be a solution. Let us now show that the number of possible solutions is exponential only with $k$, and it is thus possible to try all the possibilities.

To do so, we first give a bound to the number of children for $\mathcal{M}$. There are at most $k$ nodes in each child of $\mathcal{M}$ (otherwise, this child cannot be in a solution). In each child of $\mathcal{M}$, there are at most $2^c$ different sets of colors associated to
each node. Since there are at most $k$ nodes in each child of $M$, there are at most $(2^c)^k$ different children of $M$. A same child of $M$ cannot occurs more than $k$ times (otherwise, the next occurrences cannot be in a solution for a motif of size $k$). Therefore, there are at most $k(2^c)^k$ children to consider for $M$.

We bounded the number of children for $M$. We now choose the potential union of children of $M$ in the solution – we must choose $i$ among the $k(2^c)^k$ children, where $i$ goes from 1 to $k$. This is bounded by $(k(2^c)^k)^{k+1}$. Finally, for each union of chosen children, there are $c$ possible colors for the nodes (there are at most $k$ of them), which lead to at most $c^k$ tests. The overall complexity of the algorithm is thus exponential only in $k$. \qed