Some Results on Topological Colored Motifs in Metabolic Networks

Elói Araújo, Marco A. Stefanes

Abstract— **In this work, we address the topological colored motif search problem in metabolic networks. This problem is a concern in biology, which seeks to describe the functions and the evolution of metabolism. Recently, several variations of this problem have been studied. Here, we present some hardness results for finding motifs. Furthermore, we describe the first polynomial algorithm for the case in which the motif is a colorful tree. We also detail a data structure that allows finding all of these types of motifs in a metabolic network.**

I. Introduction

In the context of structural analysis of metabolic networks, the problem of searching occurrences of motifs plays a central role. In metabolic networks, motifs are associated with both the basic modules of molecular information and the functional and structural features. Recently, several papers have addressed the problem considering some permutation of the motif inside a specific topology of a network [11], [16], [1]. In their pioneer study Lacroix et al. [13] describe the topological colored motif problem and show that it is an NP-hard problem for the general case. They conjectured that the problem could be solved in polynomial time when the motif is a colorful tree, where a motif is *colorful* if each vertex has a color and each color appears once in the motif.

In this article, we measure the impact of several constraints over topological colored motifs, all of them considering that the motif is colorful. We show the hardness of some motif search problems. In the first one, if it is given a vertex-colored graph G, representing a metabolic network, and a colorful motif M, so we show that searching a simple subgraph of G isomorphic to M is NP-hard. We show that the motif search problem is NP-hard if the colorful motif is M as an induced subgraph, even when M is a tree. Interestingly, we also demonstrate that if M is a colorful tree, then finding M as a simple subgraph can be solved in polynomial time confirming the conjecture of Lacroix et al. [13].

We also address two enumeration problems related to motifs: the former one consists of enumerating all occurrences of a colorful motif M in a vertex-colored graph G and the other consists of finding the maximum number of disjoint occurrences of M in G. For the first, we provide a data structure in polynomial time and, for the second, it can be showed that it is NP-hard.

This work is organized as follows: in Section II, we provide some definitions and background; in Section III, we show some hardness results; in Section IV, we present a polynomial algorithm for the case when the motif is a colorful tree and we describe a data structure that allows finding all of these kind of motifs in a metabolic network; and, finally, in Section V, we draw some conclusions.

Elói Araújo and M. A. Stefanes are with Faculdade de Computação, Universidade Federal de Mato Grosso do Sul, Brazil {feloi,marco}@facom.ufms.br

978-1-4799-3163-7/13/$31.00 ©2013 IEEE

II. Definitions and Background

Formally, a *metabolic network* can be defined as a collection of objects and the relations among them [12]. *Chemical compounds* are small molecules transferred, synthesized or consumed inside an organism. A metabolic network can be modeled as a simple graph, called *graph compound*, where the vertices correspond to chemical compounds and the edges correspond to reactions where a compound is substrate and the other is a product. *Network motifs* are patterns of interconnections that occur in different parts of a metabolic network with high frequency.

A *vertex-colored graph* is a graph where each vertex has a color. This kind of graph has several applications in biological networks (Protein-protein Interaction, Metabolic and Regulatory networks) [9], [3], [15]. In the context of motifs, a metabolic network can be represented in a compact manner by a vertex-colored graph, called *reaction graph*, where vertices represent chemical reactions and two vertices are linked if and only if the reactions share a same chemical compound [13]. *Reaction Motif* is a motif in a reaction graph. Throughout this work, we assume that all colors of vertices in the reaction graph appear in the motif.

We denote the color of an object u by $\mathbf{c}(u)$. If G is a vertex-colored graph, we denote the set of vertices and the set of edges in G by $V(G)$ and $E(G)$ respectively. We write uv to represent edge $\{u, v\}$ and we say that u and v are *adjacent*. The vertex-colored graphs H and G are *isomorphic* if there is a bijection θ between the vertex set of H and G such that $uv \in E(H)$ if and only if $\theta(u)\theta(v) \in E(G)$ and $\mathbf{c}(u) = \mathbf{c}(\theta(u))$ for each $u, v \in V(H)$. We write $H \equiv G$ if H and G are isomorphic.

If G is a graph, $V' \subseteq V(G)$ and $E' \subseteq E(G)$, we denote the graph obtained removing all vertices in V' and all edges connecting any vertex in V' to another vertex from G by $G - V'$ and we denote the graph obtained removing all edges in E' from G by $G - E'$. A graph H is a *simple subgraph* of G or simply a *subgraph* of G if there are a set V' of vertices and a set E' of edges such that $H = (G - V') - E'$ and a graph H is an *induced subgraph of* G *by* $V(G) - V'$ if $H = G - V'$. We also denote $G - V'$ by $G[V(G) - V']$. We write $H \prec G$ if H is isomorphic to a subgraph of G.

Let G and M be a vertex-colored graph representing a reaction graph and a motif respectively. In order to address problems related to motifs in metabolic networks, a motif can be characterized as follows: (1) a *Topological Motif* is one represented by a reaction graph in which all its vertices have the same color; here, depending on the specific problem we are interested in answering if $M \prec G$ or if M is an induced subgraph of G. (2) a *Colored Motif* is defined as a multiset of colors. Here, there is an occurrence of a motif M in G if an injective function $f : M \rightarrow V(G)$ exists such that for each $u \in M$, $f(u) = v$ implies that u and v share some color, and the image of f induces a connected subgraph of G. (3) a *Topological Colored Motif* is a motif represented by a reaction graph; determining an occurrence of M in G in this case takes into account both connectivity and coloring. Again, like (1), we can be interested in a motif that it is a simple or an induced subgraph of a metabolic network.

A general version of a motif search problem addresses the question as to whether the motif M exists in G. All known variants of search problems concerned with topological motifs are NP-complete, even if M is a path, since it is easy to see a reduction from the Hamiltonian Path problem, which is NP-complete [8]. The restricted version of the topological motifs to an induced subgraph is known as *subgraph isomorphism problem*, which is also NP-complete since it can be reduced to a problem of deciding whether a graph contains a clique larger than a given size, which is NP-complete [8]. Search problems concerned with colored motifs showed to be NP-complete if a repetition of colors occurs [13], even if G is a tree. Most recently, it was shown [7] that search problem of colored motifs is also NP-complete even if M consists of $|M|$ distinct colors and each set of multiset M is an unitary set. Some variations of the colored motifs problems are APX-hard, even if the graph is a tree [5]. Some solutions using parameterized algorithms for searching colored motif have been provided by Dondi *et al* [6], and by Betzler *et al* [2]. Guillemot and Sikora [10] present an FPT algorithm for counting colored motifs.

III. Some Hardness Results

Considering that the motif is colorful, a decision version of the Motif Search problem can be formulated as:

Problem 1 (Simple Subgraph): Given a vertex-colored graph G and a colorful M, $M \prec G$?

Theorem 1: Simple Subgraph problem is NP-complete.

Proof: It is easy to see that it is in NP. So, we provide a reduction from the 3-SAT problem, which is known to be NP-complete [8]. Given an arbitrary Boolean formula Φ with m clauses $C_1, \ldots, C_m$ as an instance of 3-SAT, construct G with $3m$ vertices where each vertex represents a literal of a clause in Φ. Vertices u and v in G have the same color if and only if u and v come from the same clause in Φ. An edge $uv \in E(G)$ if, and only if, the literals that the vertices u and v represent are not opposite. Furthermore, M is a colorful clique with m vertices, whose colors are those in vertices in G. This transformation can be done in polynomial time. Next, we argue that the Φ is satisfiable if and only if $M \prec G$.

Suppose that Φ is satisfiable. So, there is a true assignment to the variables that satisfy all clauses. Let S be a set of vertices corresponding to m literals that belong to a different clause and that satisfy Φ. By definition of G, it follows that $G[S] \equiv M$ and therefore $M \prec G$.

On the other hand, suppose that $M \prec G$. So, there is a subgraph H of G, such that $H \equiv M$. So, H is a colorful clique. It follows that the set of vertices in H represents a set S of m not opposite literals in Φ, one per clause. Therefore, a true assignment to literals in S satisfies Φ. ■

Now, consider the following variation of the above problem: given a vertex-colored graph G and a colorful M, is there an induced subgraph of G isomorphic to M? Note that if $M \prec G$ and M is a colorful clique, there is H that is also an induced subgraph of G. So, the same reduction and arguments in the proof of Theorem 1 may be used to show that it is also an NP-complete problem.

Both biological networks and motifs can be represented by directed graphs. This is specially relevant when we are searching motifs in a gene regulatory network. Unfortunately, that version of search motif problem is not polynomial, unless P=NP, even whether the input are *directed acyclic graphs* (DAGs), since DAG isomorphism is an NP-complete problem [14]. A related version of the motif search problem is as follows: given two vertex-colored DAGs G and M such that each color appears only once in M, is there a subgraph of G isomorphic to M? Using some adjustments the result of Theorem 1 can be extended to show that this version of the problem is also NP-complete.

Another variation of the problem is given below.

Problem 2 (Induced Tree): Given a vertex-colored graph G and a colorful tree M, is there an induced subgraph of G isomorphic to M?

Theorem 2: Induced Tree problem is NP-complete.

Proof: It is easy to see that this problem is in NP and, as well as in Theorem 1, we present a reduction from the 3-SAT problem. So, given an arbitrary Boolean formula Φ with m clauses $C_1, \cdots C_m$ as an instance of 3-SAT, construct G with $3m + 1$ vertices where each of $3m$ first vertices represents a literal of a clause in Φ. We call the extra vertex *core*. Vertices u and v in G have the same color if and only if u and v come from the same clause in Φ. The core vertex has a different color from any other vertex and it is linked to each one of the $3m$ remaining vertices. Moreover, two of the vertices representing literals are linked if and only if the literals they represent are opposite. Furthermore, M is a colorful star with $m + 1$ vertices, whose colors are those in $V(G)$ such that the color of its center is the color of the core. See Figure 1. This transformation can be done in polynomial time.

Now, we show that the formula Φ is satisfiable if and only if there is an induced subgraph of G isomorphic to star M. Suppose that Φ is satisfiable. So, there is a true assignment to the variables that satisfies all clauses. Let S be a set of vertices corresponding to m literals that belong to a different clause and that satisfy Φ. By definition of G, it follows that $G[S \cup \{core\}] \equiv M$ and therefore $G[S \cup \{core\}]$ is an induced subgraph of G isomorphic to M.

Now, suppose that there is an induced subgraph H of G that is isomorphic to M. Because the core color is unique

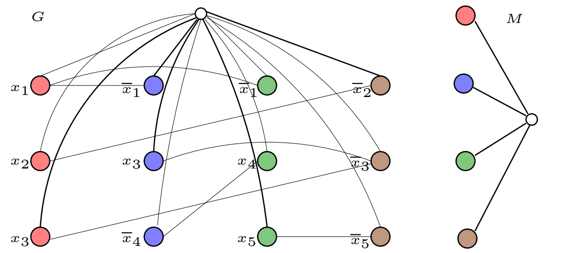

Fig. 1. Example of graphs G and M constructed from formula $C_1 \wedge C_2 \wedge C_3 \wedge C_4$ where $C_1 = x_1 \vee x_2 \vee x_3$, $C_2 = \overline{x}_1 \vee x_3 \vee \overline{x}_4$, $C_3 = \overline{x}_1 \vee x_4 \vee x_5$, $C_4 = \overline{x}_2 \vee \overline{x}_3 \vee \overline{x}_5$.

in G and M, the color of the center of H has the same color as the vertex core in M and the set of vertices in $V(H) - \{core\}$ represents a set S of m not opposite literals in Φ, one per clause. Therefore, a true assignment to literals in S satisfies Φ. ∎

An important optimization problem related to motifs which interests biologists is finding the maximum number of disjoint occurrences of the motif. Next, we present the decision version of this problem which can be showed that it is an NP-complete problem even when the motif is a colorful tree. The proof is omitted due to lack of space.

Problem 3 (Disjoint Occurrences): $(\mathrm{DO}(G, M, \kappa))$. Given a vertex-colored graph G, a colorful tree M and an integer κ, is there κ disjoint occurrences of M in G?

Theorem 3: Disjoint Occurrences is NP-complete.

IV. Algorithms

We begin by providing some additional definitions that we shall use in the remaining section. If the set of vertices in G is empty, we write $G = \emptyset$. If v and w are vertices and $v \in G$ but $w \notin G$, we denote the graph G adding vertex w and edge vw by $G + \{vw\}$. If $\mathtt{C}$ is a set of colors, we write the induced graph of T by vertices whose colors are in $\mathtt{C}$ as $T[\mathtt{C}]$.

Problem 4 ($\mathbf{TCG}$): Tree in a colored graph. Given a vertex-colored graph G and a colorful tree T, determine if $T \prec G$, i.e.

$$\mathbf{TCG}(G, T) = \begin{cases} \mathtt{yes} & \text{if } T \prec G, \\ \mathtt{no} & \text{otherwise.} \end{cases}$$

Algorithm $\mathbf{TCG}(G, T)$

1. if $|T| = 0$, return `yes`;
2. if $|T| = 1$, let $v \in V(T)$
2.1 if $\exists w \in V(G) : \mathbf{c}(w) = \mathbf{c}(v)$, then return `yes`;
2.2 else return `no`;
3. if $|T| > 1$, then
3.1 let $v, v' \in V(T) : v$ is a leaf and $vv' \in E(T)$;
$\mathcal{A} = \{u \in V(G) : \mathbf{c}(u) = \mathbf{c}(v)\}$;
$\mathcal{B} = \{u \in V(G) : \mathbf{c}(u) = \mathbf{c}(v')$ and u does not have neighbors whose color is $\mathbf{c}(v)\}$;
3.2 $G' := G - (\mathcal{A} \cup \mathcal{B})$;
3.3 $T' := T - \{v\}$;
3.4 return $\mathbf{TCG}(G', T')$.

Correctness and Complexity

It is clear that this algorithm is correct when $|T| \leq 1$. Then, for $|T| > 1$, we must show that $\mathbf{TCG}(G, T) = \mathbf{TCG}(G', T')$. The theorem below states this formally.

Theorem 4: Let T and G be a colorful tree with $n > 1$ vertices and a vertex-colored graph respectively that are given as an input to $\mathbf{TCG}$. Consider that G' and T' are vertex-colored graphs computed by the lines 3.2 and 3.3 respectively in $\mathbf{TCG}$. So, $T \prec G$ if and only if $T' \prec G'$.

Proof: Let $v, v' \in V(T)$ such that v is a leaf and $vv' \in E(T)$.

First suppose that $T \prec G$. Let $\overline{G}$ be a subgraph of G, such that $\overline{G} \equiv T$. Because T is colorful and $\overline{G} \equiv T$, there are $w, w' \in V(\overline{G})$ such that $\mathbf{c}(w) = \mathbf{c}(v)$ and $\mathbf{c}(w') = \mathbf{c}(v')$ and $w'w \in V(\overline{G})$. Note that $T' = T - \{v\} \equiv \overline{G} - \{w\} = \overline{G}'$. Furthermore, since $\overline{G}' = \overline{G} - \{w\}$ is a subgraph of G', it follows that $T' \prec G'$.

Conversely, suppose that $T' \prec G'$ and $T' = T - \{v\}$. Let $\overline{G}'$ be a subgraph of G', such that $\overline{G'} \equiv T'$. There is a vertex w' in $\overline{G}'$ such that $\mathbf{c}(w') = \mathbf{c}(v')$ and, by the algorithm, it must be a neighbor to a vertex w in G, with color $\mathbf{c}(v)$. So, $\overline{G} = \overline{G}' + \{w'w\}$ is a subgraph of G. Also, it follows that $T \equiv \overline{G}$. Therefore, $T \prec G$. ∎

Now, let n_t be the number of vertices of T, and n_g, m_g be the number of vertices and edges of G respectively. Since the vertices and edges in $\mathcal{A} \cup \mathcal{B}$ can be removed from G in $O(n_g + m_g)$ time, our algorithm spends total time

$$\mathcal{T}(n_t) = \begin{cases} 1 & \text{if } n_t \leq 1, \\ \mathcal{T}(n_t - 1) + O(n_g + m_g) & \text{otherwise.} \end{cases}$$

So, the algorithm spends $O(n_t(n_g + m_g))$ time.

Maximum clean graphs

In general, we are not only interested in whether there is a subgraph of G that is isomorphic to T but we also want to find such a subgraph. We would also like to find several or even all different subgraphs of G that are isomorphic to T.

The number of subgraphs of G that are isomorphic to the tree T can be very large compared to the input size, which makes it prohibitive to find all isomorphic subgraphs of T at least in the general case. Furthermore, it is useful to have a simpler structure than G from which it is possible to obtain all possible subgraphs of G that are isomorphic to T. This structure is described below.

We say that a graph H is a *clean graph* from T if each vertex in H is a vertex of some subgraph of H that is isomorphic to T and each edge in H is an edge of some subgraph of H that is isomorphic to T. For a vertex-colored graph G, we say that H is the *maximum clean graph from T of G* if H is a clean graph from T and any subgraph of G which is isomorphic to T is also a subgraph of H. The algorithm $\mathbf{MCG}$ below finds the maximum clean graph from T of G.

Correctness and Complexity

In order to prove that H is the maximum clean graph from T of G, since vertices and edges removed from H by the

Algorithm $\mathbf{MCG}(G, T)$

1. $H := G$;
2. for each $v \in V(H)$
   2.1 if $\mathbf{c}(v)$ is not in T, then $H := H - \{v\}$;
3. for each $vv' \in E(H)$
   3.1 if there is no edge ww' in T such that $\mathbf{c}(v) = \mathbf{c}(w)$ and $\mathbf{c}(v') = \mathbf{c}(w')$, then $H := H - \{vv'\}$;
4. while there are $v \in V(H)$ and $w, w' \in V(T)$ such that $\mathbf{c}(v) = \mathbf{c}(w)$, $ww' \in E(T)$ but there is no $v' \in V(H)$ such that $vv' \in E(H)$ and $\mathbf{c}(v') = \mathbf{c}(w')$, do
   4.1 $H := H - \{v\}$;
5. return H.

algorithm cannot be vertices and edges of subgraphs of G that are isomorphic to T, it is enough to show that H is a clean graph from T. We do this through lemmas 1 and 2. But first, we can easily verify the following auxiliary result.

Fact 1: Let H be a vertex-colored graph obtained by algorithm $\mathbf{MCG}$ which receives tree T as entry, $v \in V(T)$ be a leaf and $\mathcal{A} = \{w \in V(H) : \mathbf{c}(w) = \mathbf{c}(v)\}$. Then, $\mathbf{MCG}(H, T - \{v\}) = H - \mathcal{A}$.

Lemma 1: Let H be a vertex-colored graph obtained by algorithm $\mathbf{MCG}$ which has tree T as entry. Then, any arbitrary vertex x in H is a vertex of some subgraph $\overline{G}$ in H that is isomorphic to T.

Proof: Let n be a number of vertices of T. We prove the lemma by induction of n. Suppose that $n \leq 1$. If $n = 0$, after step 1, H has no vertex and the proof is done by vacuous truth. If $n = 1$, after step 1, the proof is also done because each vertex in H is isomorphic to T. Assume then $n > 1$.

If $n > 1$, then we have that T has two different leaves. Since T is a colorful tree, it follows that, for one of these leaves, say v, we have $\mathbf{c}(x) \neq \mathbf{c}(v)$. Let w be the neighbor of v in T.

From Fact 1, we have that $\mathbf{MCG}(H, T - \{v\}) = H - \mathcal{A}$ where $\mathcal{A} = \{u : \mathbf{c}(u) = \mathbf{c}(v)\}$. Since $\mathbf{c}(x) \neq \mathbf{c}(v)$, we have that $x \in V(H - \mathcal{A})$. By the induction hypothesis, x is a vertex of a subgraph $\overline{G}' \equiv T - \{v\}$ in $H - \mathcal{A}$. Let $w' \in \overline{G}'$ such that $\mathbf{c}(w') = \mathbf{c}(w)$. By $\mathbf{MCG}(G, T)$, w' must have a neighbor v' whose color is $\mathbf{c}(v)$. It follows that $\overline{G} = \overline{G}' + \{w'v'\} \equiv T$, $\overline{G}$ is a subgraph of H and $x \in V(\overline{G})$. ■

Lemma 2: Let H be a vertex-colored graph obtained by algorithm $\mathbf{MCG}$ which has tree T as entry. Then, any arbitrary edge xy in H is an edge of some subgraph $\overline{G}$ in H that is isomorphic to T.

Proof: From Lemma 1 there are subgraphs $\overline{G}_x \equiv T$ and $\overline{G}_y \equiv T$ in H which contain vertices x and y respectively. It follows that there are vertices y' and x' whose colors are $\mathbf{c}(y)$ and $\mathbf{c}(x)$ respectively and xy' is an edge in $\overline{G}_x$ and $x'y$ is an edge in $\overline{G}_y$. Let G'_x the component of $\overline{G}_x - \{xy'\}$ that has vertex x and G'_y the component of $\overline{G}_y - \{x'y\}$ that has vertex y. The graph $\overline{G}$ defined as the components G'_x and G'_y adding edge xy is a subgraph of H isomorphic to T. ■

In order to analyze the time complexity of $\mathbf{MCG}$, we consider that G and T are represented by adjacency matrices and the set of colors by an ordered array. Let n be the number of vertices of the entry of $\mathbf{MCG}$. In this case, Step 1 can be implemented in $O(n^2)$ time because H is a copy of G and adjacency matrix G has $O(n^2)$ entries; the color of each vertex in H can be checked in $O(1)$ time and be removed in $O(n)$ time, which implies that Step 2 can be performed by spending $n \cdot (O(1) + O(n)) = O(n^2)$ time; we can verify if the colors of the vertices of an edge in H are also the colors of vertices of an edge in T in $O(1)$ time, and, if necessary, its remotion can also be done in $O(1)$ time which implies, since we have $O(n^2)$ edges, that Step 3 can be performed by spending $O(n^2) \cdot (O(1) + O(1)) = O(n^2)$ time; in Step 4, deciding on whether a vertex should be removed or not can be made spending $O(n^2)$ time, and the removal can be done in $O(n)$ time if this is the case, implying that the total time spent on Step 4 is $n \cdot (O(n^2) + O(n)) = O(n^3)$. So, $\mathbf{MCG}$ spends $O(n^3)$ time.

Finding a subgraph of H isomorphic to T

In the previous section, we showed how to find the maximum clean graph H from T of G. A simple greedy algorithm finds a subgraph $\overline{G} \equiv T$ from a clean graph H. All we need to compute $\overline{G}$ is H, i.e., tree T is implicit in this algorithm. Here, we assume that $H \neq \emptyset$.

Algorithm $\mathbf{Greedy}(H)$

1. $V_1 = \{x_1\}$, where x_1 is an arbitrary vertex in H;
2. $E_1 = \emptyset$;
3. $\mathtt{C}_1 := \{\mathbf{c}(x_1)\}$;
4. for $i := 2$ to n do
   4.1 choose $vx_i \in E(H)$ such that $v \in V_{i-1}$, $x_i \notin V_{i-1}$ and $\mathbf{c}(x_i) \notin \mathtt{C}_{i-1}$; (if there are multiple edges with the same property, any of them may be picked)
   4.2 $V_i := V_{i-1} \cup \{x_i\}$, $\mathtt{C}_i := \mathtt{C}_{i-1} \cup \{\mathbf{c}(x_i)\}$;
5. return $H[V_n]$.

The total time spent by $\mathbf{Greedy}$ depends on how much time is spent to find an edge vx_i in Step 4.1. If H is represented by an adjacency matrix, and m is the number of edges in H, then Step 4.2 can be executed in $O(m)$ time. As Step 4 is executed $n - 1$ times, $\mathbf{Greedy}$ algorithm may be performed in $(n-1)O(m) = O(nm)$.

In order to prove that $\mathbf{Greedy}$ works correctly, we prove the following result.

Lemma 3: $H[V_n] \equiv T[\mathtt{C}_n] \equiv T$.

Proof: We show by induction in n. Case $n = 1$, the lemma holds because Step 2 of $\mathbf{MCG}$ guarantees that the color of each vertex in H appears in T.

So, assume that $H[V_{n-1}] \equiv T[\mathtt{C}_{n-1}]$. Since T is a colorful tree with n vertices, it follows that there are $u, w \in V(T)$ such that $\mathbf{c}(u) \notin \mathtt{C}_{n-1}$, $\mathbf{c}(w) \in \mathtt{C}_{n-1}$ and $uw \in E(T)$. Since $\mathbf{c}(w) \in \mathtt{C}_{n-1}$, there is $v \in V_{n-1}$ such that v is a vertex in H and $\mathbf{c}(w) = \mathbf{c}(v)$, and Step 4 of algorithm $\mathbf{MCG}$ guarantees that there is a $x_n \notin V_{n-1}$, such that x_n is a vertex in H, $\mathbf{c}(x_n) = \mathbf{c}(u)$ and vx_n is edge in H. So, $H[V_n] \equiv T[\mathtt{C}_n]$. ■

Finding all subgraphs of H isomorphic to T

As well as in the previous section, the next algorithm has just the maximum clean graph H as the entry. Tree T is implicit. If $H = \emptyset$, then there is no subgraph of H isomorphic to T (except if $T = \emptyset$ too). So, we assume that H has at least one vertex.

Algorithm $\textbf{All\_Isomorphic}(H)$

1. if $|\{\mathbf{c}(u) : u \in V(H)\}| = 1$, then $\mathcal{G} := V$;
2. else let α be a color of a vertex of H whose neighbors have all the same color β;

2.1 $\mathcal{A} := \{u \in V(H) : \mathbf{c}(u) = \alpha\}$;

2.2 $\mathcal{G} := \emptyset$;

2.3 $\mathcal{H} := \textbf{All\_Isomorphic}(H - \mathcal{A})$;

2.4 for each vertex $v \in \mathcal{A}$ do
 $\mathcal{B} := \{u \in V(H) : \mathbf{c}(u) = \beta \text{ and } vu \in E(H)\}$;
 for each $\overline{G} \in \mathcal{H}$ such that $w \in \mathcal{B} \cap V(\overline{G})$ do
 $\mathcal{G} := \mathcal{G} \cup \{\overline{G} + \{vw\}\}$

3. return $\mathcal{G}$

Since H is a clean graph from T, the following fact holds.

Fact 2: Let T be a colorful tree such that $|V(T)| = 1$. Then, for each $v \in V(H)$, we have $v \in \mathcal{G}$, v is a subgraph of H and $v \equiv T$.

The following result shows that $\textbf{All\_Isomorphic}(H)$ returns the set of all subgraphs of H isomorphic to T.

Theorem 5: Let G be a vertex-colored graph and T be a colorful tree. Suppose that $H := \mathbf{MCG}(G, T)$, $\mathcal{G} := \textbf{All\_Isomorphic}(H)$ and $\overline{G}'$ is a subgraph of H. Then, $\overline{G}' \in \mathcal{G}$ if and only if $\overline{G}' \equiv T$.

Proof: Let n be the number of vertices in T. If $n = 1$, then, since $H = \mathbf{MCG}(G, T)$, all vertices in H have the same color. So, from Step 1 of $\textbf{All\_Isomorphic}$, we have that $v \in \mathcal{G}$ for each $v \in V(H)$. On the other hands, from Fact 2, $v \equiv T$ for each $v \in V(H)$. So, the theorem holds if $n = 1$. Then, we assume that $n > 1$.

Suppose that $\overline{G}' \in \mathcal{G}$. By Step 2.4 we have $\overline{G}' = \overline{G} + \{vw\}$ where $w \in \mathcal{B}$ is a vertex in $\overline{G} \in \mathcal{H}$ and $v \in \mathcal{A}$. Let $u, u' \in V(T)$ such that $\mathbf{c}(u) = \mathbf{c}(v)$ and $\mathbf{c}(u') = \mathbf{c}(w)$ and note that, because H is a clean graph from T, it follows that u and u' are neighbors in T. Because H is the maximum clean graph from T and every neighbor of a vertex of color $\mathbf{c}(u)$ has color $\mathbf{c}(u')$, it follows that u is a leaf in T. It follows from Fact 1 that $H - \mathcal{A}$ is the maximum clean graph for $T - \{u\}$. By the induction hypothesis, it follows that $\overline{G}$ is a subgraph of $H - \mathcal{A}$ isomorphic to $T - \{u\}$. It follows that $\overline{G}' = \overline{G} + \{vw\}$ is isomorphic to $T = T - \{u\} + \{uu'\}$. So, for $n > 1$, if $\overline{G}' \in \mathcal{G}$, then $\overline{G}'$ is isomorphic to T.

Conversely, suppose that $\overline{G}'$ is a subgraph of H isomorphic to T. Let v and w be vertices in $\overline{G}'$. By the induction hypothesis, $\overline{G} := \overline{G}' - \{v\} \in \textbf{All\_Isomorphic}(H - \mathcal{A})$. Then, Step 2.4 of $\textbf{All\_Isomorphic}$ guarantees that $\overline{G}' = \overline{G}' + \{vw\} \in \mathcal{G}$. ■

V. CONCLUSIONS AND FUTURE WORKS

In this article, we solved an open problem from literature by describing a polynomial algorithm for finding colorful tree motifs in a vertex-colored graph. We also gave a data structure that allows counting or enumerating all motifs we are interested in a network. We further demonstrated some hardness results concerning topological colored motifs. Since the main goal for searching motifs is to help in the structural analysis of biological networks, it would be interesting to make an experimental study of these algorithms in actual biological networks. Previous works[16], [4] have showed the validation of the model. Some improvement can be obtained using mixed techniques such as the fixed parameter approach or approximated algorithms. A variation of motif search problem has been studied considering an edge-weighted motif, we would like to address this variation as a forthcoming work.

VI. ACKNOWLEDGMENTS

The authors would like to thank Yoshiko Wakabayashi, Marie-France Sagot and Estela M. Rodrigues for their valuable discussions.

REFERENCES

[1] Abhimanyu M Ambalath, Radheshyam Balasundaram, Chintan Rao, Venkata Koppula, Neeldhara Misra, Geevarghese Philip, and MS Ramanujan. On the kernelization complexity of colorful motifs. In *Parameterized and Exact Computation*, pages 14–25. Springer, 2010.

[2] Nadja Betzler, Michael R Fellows, Christian Komusiewicz, and Rolf Niedermeier. Parameterized algorithms and hardness results for some graph motif problems. In *Comb. Pattern Match.*, pages 31–43, 2008.

[3] Hans L Bodlaender, Michael R Fellows, Michael A Langston, Mark A Ragan, Frances A Rosamond, and Mark Weyer. Quadratic kernelization for convex recoloring of trees. In *Computing and Combinatorics*, pages 86–96. Springer, 2007.

[4] Sharon Bruckner, Falk Huffner, Richard M. Karp, Ron Shamir, and Roded Sharan. Topology-free querying of protein interaction networks. *Journal of Computacional Biology*, 17(3):237–252, 2010.

[5] Riccardo Dondi, Guillaume Fertin, and Stéphane Vialette. Maximum motif problem in vertex-colored graphs. In *Comb. Pattern Match.*, pages 221–235. Springer, 2009.

[6] Riccardo Dondi, Guillaume Fertin, and Stphane Vialette. Finding approximate and constrained motifs in graphs. *Theoretical Computer Science*, 483:10 – 21, 2013.

[7] Michael R Fellows, Guillaume Fertin, Danny Hermelin, and Stéphane Vialette. Sharp tractability borderlines for finding connected motifs in vertex-colored graphs. In *Automata, Languages and Programming*, pages 340–351. Springer, 2007.

[8] Michael R Garey and David S Johnson. *Computers and intractability*, volume 174. Freeman San Francisco, CA, 1979.

[9] M. C. Golumbic, H. Kaplan, and R. Shamir. On the complexity of DNA physical mapping. *Advances in Appl. Math.*, 15:251–261, 1994.

[10] Sylvain Guillemot and Florian Sikora. Finding and counting vertex-colored subtrees. *Algorithmica*, 65(4):828–844, 2013.

[11] Michel Koskas, Gilles Grasseau, E Birmelé, Sophie Schbath, and Stéphane Robin. Nemo: Fast count of network motifs. *Book of Abstracts for Journées Ouvertes Biologie Informatique Mathématiques (JOBIM)*, pages 53–60, 2011.

[12] Vincent Lacroix, Ludovic Cottret, Patricia Thébault, and M-F Sagot. An introduction to metabolic networks and their structural analysis. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 5(4):594–617, 2008.

[13] Vincent Lacroix, Cristina G Fernandes, and M-F Sagot. Motif search in graphs: application to metabolic networks. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 3(4):360–368, 2006.

[14] D. Maier. The complexity of some problems on subsequences and supersequences. *Journal of the ACM (JACM)*, 25(2):322–336, 1978.

[15] Shlomo Moran and Sagi Snir. Convex recolorings of strings and trees: Definitions, hardness results and algorithms. In *Algorithms and Data Structures*, pages 218–232. Springer, 2005.

[16] Sophie Schbath, Vincent Lacroix, and Marie-France Sagot. Assessing the exceptionality of coloured motifs in networks. *EURASIP Journal on Bioinformatics and Systems Biology*, 2009:3, 2009.