Parameterized Complexity and Approximation Issues for the Colorful Components Problems

Riccardo Dondi[1], Florian Sikora[2]

[1] Dipartimento di Scienze umane e sociali, Università degli Studi di Bergamo, ITALY
riccardo.dondi@unibg.it
[2] Université Paris-Dauphine, PSL, CNRS UMR 7243, LAMSADE, FRANCE
florian.sikora@dauphine.fr

Abstract. The quest for colorful components (connected components where each color is associated with at most one vertex) inside a vertex-colored graph has been widely considered in the last ten years. Here we consider two variants, Minimum Colorful Components (MCC) and Maximum Edges in transitive Closure (MEC), introduced in the context of orthology gene identification in bioinformatics. The input of both MCC and MEC is a vertex-colored graph. MCC asks for the removal of a subset of edges, so that the resulting graph is partitioned in the minimum number of colorful connected components; MEC asks for the removal of a subset of edges, so that the resulting graph is partitioned in colorful connected components and the number of edges in the transitive closure of such a graph is maximized. We study the parameterized and approximation complexity of MCC and MEC, for general and restricted instances.

1 Introduction

The quest for colorful components inside a vertex colored graph has been a widely investigated problem in the last years, with application for example in bioinformatics [12,5,8]. Roughly speaking, given a vertex-colored graph, the problem asks to find the colorful components of the graph, that is connected components that contain at most one vertex of each color. While most of the approaches have focused on the identification of a single connected colorful component, the identification of the minimum number of colorful connected components that match a given motif has been considered in [4,7].

Here we consider a similar framework, where instead of looking for a single colorful component inside a vertex-colored graph, we ask for a partition of the graph vertices in colorful components. This approach stems from a problem in bioinformatics, and more specifically in comparative genomics. In this context, a fundamental task is to infer the relations between genes in different genomes and, more precisely, to infer which genes are orthologous, that is those genes that originate via a speciation event from a gene of an ancestral genome.

A graph approach has been proposed aiming to identify disjoint orthology sets, where each of such sets corresponds to colorful disjoint component in the given graph [13].

Different combinatorial problem formulations, based on different objective functions, have been proposed and studied in this direction [13,2]. Here, we considered two such approaches, Minimum Colorful Components (MCC) and Maximum Edges in transitive Closure (MEC). Given a vertex-colored graph, both problems ask for the removal of some edges so that the resulting graph is partitioned in colorful components but with different objective functions. The former aims to minimize the number of connected colorful components, while the latter aims to maximize the transitive closure of the resulting graph. A related but different problem has been considered in [5], where the objective function is the minimization of edge removal, so that the computed graph consists only of colorful components.

Previous Results. Given a graph on n vertices, MCC is known not only to be NP-hard, but also not approximable within factor $O(n^{1/14-\varepsilon})$ unless P=NP [2]. It is easy to see that the reduction leading to this inapproximability result implies also that MCC cannot be solved in time $n^{f(k)}$ for any function f, where k is the number of colorful components.

MEC is known to be APX-hard even when colored by at most three colors (while it is solvable in polynomial time for two colors), and, unless P=NP, it is not approximable within factor $O(n^{1/3-\varepsilon})$ when the number of colors is arbitrary, even when the input graph is a tree where each color appears at most twice [1]. A heuristic to solve MEC is presented in [13], while in [1], the authors present a polynomial-time $\sqrt{2 \cdot OPT}$ approximation algorithm.

Contributions and organization of the paper. In this paper we investigate more deeply the complexity of MCC and MEC. More precisely, we show in Section 3 that MCC on trees is essentially equivalent to Minimum MultiCut on Trees, thus MCC is not approximable within factor $1.36 - \varepsilon$ unless P=NP for any $\varepsilon > 0$, but 2-approximable, it is fixed-parameter tractable and it admits a poly-kernel (when the parameter is the number of colorful components). Moreover, in Section 4 we show that MCC is easily solvable in polynomial time on paths, while it is not in XP class when parameterized by the structural parameter Distance to Disjoint Paths.

Then we consider the parameterized complexity of MEC with respect to the number k of edges in the transitive closure of a solution. For this parameter we give in Section 5 a parameterized algorithm, by reducing the problem to an exponential kernel. We use a similar idea in Section 6, to improve it to a poly-kernel for MEC when the input graph is a tree. Finally, we show in Section 7 that results similar to those of Section 4, hold also for MEC.

2 Definitions

In this section we introduce some preliminary definitions. Consider a set of colors $C = \{c_1, \ldots, c_q\}$. A C–colored graph $G = (V, E, C)$ is a graph where every vertex in V is associated with a color in C; the color associated with a vertex $v \in V$

is denoted by $c(v)$. A connected component induced by a vertex set $V' \subseteq V$ is called *a colorful component*, if it does not contain two vertices having the same color. If a graph has t connected components where each component $i \in [t]$ has exactly n_i vertices, the number of edges in its transitive closure is defined by $\sum_{i=1}^{t} \frac{n_i(n_i-1)}{2}$.

Next, we introduce the formal definitions of the optimization problems we deal with.

MINIMUM COLORFUL COMPONENTS (MCC)

• **Input**: a C-colored graph $G = (V, E, C)$.
• **Output**: remove a set of edges $E' \subseteq E$ such that each connected component in $G' = (V, E \setminus E', C)$ is colorful, and the number of connected components of G' is minimized.

MAXIMUM EDGES IN TRANSITIVE CLOSURE (MEC)

• **Input**: a C-colored graph $G = (V, E, C)$.
• **Output**: remove a set of edges $E' \subseteq E$ such that each connected component in $G' = (V, E \setminus E', C)$ is colorful, and the number of edges in the transitive closure of G' is maximum.

The parameterized versions of MCC and MEC are defined analogously (and abusively denoted with the same names), with the addition in the input of an integer k, that denotes the number of connected components in G' for MCC and the number of edges in the transitive closure of G' for MEC.

Notice that, when considering an instance of MCC and MEC, we assume that E contains no edge $\{u, v\}$ with $c(u) = c(v)$, otherwise such an edge can be deleted from E as u and v will not be part of the same colorful component in any feasible solution of MCC or MEC.

Complexity A parameterized problem (I, k) is said *fixed-parameter tractable* (or in the class FPT) with respect to a parameter k if it can be solved in $f(k) \cdot |I|^c$ time (in *fpt-time*), where f is any computable function and c is a constant (see [9] for more details about fixed-parameter tractability). The class XP contains problems solvable in time $|I|^{f(k)}$, where f is an unrestricted function.

A powerful technique to design parameterized algorithms is *kernelization*. In short, kernelization is a polynomial-time self-reduction algorithm that takes an instance (I, k) of a parameterized problem P as input and computes an equivalent instance (I', k') of P such that $|I'| \leqslant h(k)$ for some computable function h and $k' \leqslant k$. The instance (I', k') is called a *kernel* in this case. If the function h is polynomial, we say that (I', k') is a polynomial kernel.

Concerning approximation definitions, we refer the reader to some reference textbook like [3].

3 MCC for Trees: Parameterized Complexity and Approximability

In this section, we show that MCC on trees is essentially equivalent to the Minimum Multi-CUT problem on Trees (M-CUT-T), thus the positive and negative results of (M-CUT-T) for parameterized complexity and approximability transfer to MCC. We recall that M-CUT-T, given a tree T_M and a set S_M of pairs of terminals, asks if there exist a minimum cut (that is a set of removed edges) such that, for each pair $(x, y) \in S_M$, x and y are disconnected through that cut.

3.1 Positive results

We show that MCC on trees admits an FPT algorithm (and a polynomial kernel) and a 2-approximation algorithm by reducing MCC to M-CUT-T. We first describe the reduction. Given a colored tree $G_T = (V, E, C)$ as an instance of MCC, we define an instance (T_M, S_M) of M-CUT-T as follows: T_M is exactly G_T (except for the colors of the vertices); for each pair (x, y) of vertices in G_T such that $c(x) = c(y)$, we define a pair (x, y) in S_M.

Now, we prove the main lemma of this section.

Lemma 1. *Consider an instance G_T of MCC and the corresponding instance (T_M, S_M) of M-CUT-T. Then: (1) given a solution of MCC on G_T consisting of $k+1$ connected components, a solution of M-CUT-T on (T_M, S_M) consisting of k edges cut can be computed in polynomial time; (2) given a solution of M-CUT-T on (T_M, S_M) consisting of k edges, a solution of MCC on G_T consisting of $k+1$ connected components can be computed in polynomial time.*

Proof. Consider a solution of MCC consisting of $k+1$ components obtained by removing a set E' of k edges. Then, E' is a solution of M-CUT-T over instance (T_M, S_M). Indeed, for each pair $(x, y) \in S_M$, $c(x) = c(y)$, hence the two vertices belong to different connected components after the removal of edges in E'.

Conversely, consider a solution E' of M-CUT-T over instance (T_M, S_M), with $|E'| = k$. Then, remove the edges in E' from G_T and consider the $k+1$ connected components induced by this removal in G_T. Since each pair $(x, y) \in S_M$ is disconnected after the removal of E', it follows that each connected component of G_T after the removal of E' is colorful. □

We can now easily give the main result of this section:

Theorem 2. *If the input graph of MCC is a tree, MCC can be solved in time $O^*(1.554^k)$[3] where k is the natural parameter and also admits a 2-approximation algorithm.*

Lemma 1 implies also a poly-kernel for MCC on trees.

Theorem 3. *If the input graph of MCC is a tree, it is possible to compute in polynomial time a kernel of size $O(k^3)$ where k is the natural parameter.*

[3] The O^* notation suppresses polynomial factors.

3.2 Approximation lower bound of MCC on trees

Let us now prove a lower bound for the approximation of MCC on trees, by giving a reduction from M-CUT-T. Starting from an instance (T_M, S_M) of M-CUT-T, we compute a colored tree $G_T = (V, E, C)$, input of MCC, as follows. First, G_T is isomorphic to T_M, and we color each vertex v of G_T as c_v. Denote by E_1 the edge set of such a tree. Then for each pair $(u, v) \in S_M$, we define a leaf u_v adjacent to v and colored $c_{u,v}$ and a leaf v_u adjacent to u and colored $c_{u,v}$ (see Figure 1). Denote by E_2 the edge set introduced by adding these edges.

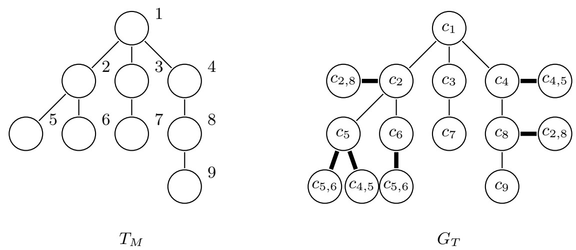

Fig. 1. Sample construction of G_T from T_M with $S_M = \{(2, 8), (5, 6), (4, 5)\}$. Edge set E_2 of T is drawn thick. For ease, colors of G_T are drawn inside the nodes. On possible solution for this instance of M-CUT-T cuts edges $\{\{2, 6\}, \{1, 4\}\}$ and implies 3 colorful connected components in the corresponding instance of MCC.

Lemma 4. *Given a solution of MCC on $G_T = (V, E, C)$ consisting of k colorful components, we can compute in polynomial time a solution of MCC on $G_T = (V, E, C)$ consisting of at most k colorful components such that the edges cut belong only to set E_1.*

Proof. Consider the case that an edge $\{u, v\}$ has been deleted, where v is a leaf introduced in G_T. Then, notice that the removal of edge $\{u, v\}$ makes v an isolated vertex. By construction u and v (and each leaf adjacent to u) have different colors. Hence there are two possible cases: either the colorful component H that contains u does not include vertices colored by c_v, hence we can add v to H, thus we can avoid removing edge $\{u, v\}$, or there is a vertex w colored by c_v in H. In this case we can remove an edge of E_1, which separates w from u without removing edge $\{u, v\}$; such an edge must exist, since v and w are leaves incident in different internal vertices. □

Lemma 5. *Consider an instance (T_M, S_M) of M-CUT-T and the corresponding instance $G_T = (V, E, C)$ of MCC. Then: (1) given a solution of M-CUT-T over instance (T_M, S_M) that cuts k edges, we can compute in polynomial time a solution of MCC over instance $G_T = (V, E, C)$ consisting of at most $k + 1$ colorful components; (2) given a solution of MCC over instance $G_T = (V, E, C)$*

consisting of at most $k+1$ colorful components, we can compute in polynomial time a solution of M-CUT-T over instance (T_M, S_M) that cuts at most k edges.

Since M-CUT-T cannot be approximated within factor 1.36 (since it is as hard as Minimum Vertex Cover to approximate [10]), Lemma 4 and Lemma 5 allow to extend the result to MCC.

Theorem 6. *MCC on trees cannot be approximated within factor $1.36 - \varepsilon$, for any constant $\varepsilon > 0$ unless P=NP.*

4 Structural parameterization of MCC

Since the MCC problem is already hard on trees, we consider in this section the complexity of MCC when the input graph is a path or is close to a set of disjoint paths. We show that MCC can be easily solved in polynomial time, while, as a sharp contrast, MCC is not in the class XP for parameter distance to disjoint paths.

Theorem 7. *MCC on paths can be solved in $O(n^3)$-time.*

Proof. Assume that the input graph is a path $G_P = (V, E, C)$, and assume that the vertices on the path are ordered from v_1 to v_n. Define $M[j]$ as the minimum number of colorful components of a solution of MCC over instance G_P restricted to vertices $\{v_1, \dots, v_j\}$. $M[j]$, with $j > 1$, can be computed as follows:

$$M[j] = \min_{0 \leqslant t < j} M[t] + 1, \text{ such that } v_{t+1}, \dots, v_j \text{ induce a colorful component.}$$

In the base cases, it holds $M[1] = 1$, and $M[0] = 0$. Next, we prove the correctness of the dynamic programming recurrence.

We claim that given a path $G_P = (V, E, C)$ instance of MCC, there exists a solution of MCC on instance G_P restricted to vertices $\{v_1, \dots, v_j\}$ consisting of h colorful components if and only if $M[j] = h$.

It is then easy to see that the value of an optimal solution of MCC on path $G_P = (V, E, C)$ is stored in $M[n]$. Since the table $M[j]$ consists of n entries and each entry can be computed in time $O(n^2)$, it follows that MCC on paths can be computed in time $O(n^3)$. □

Let us now prove that MCC is not in XP when parameterized by the *Distance to Disjoint Paths* number d (the minimum number of vertices to remove from the input graph to have disjoint paths), even when the input graph is a tree. We prove this result by giving a reduction from Minimum Vertex Cover (MinVC) to MCC on trees.

Consider an instance $G = (V, E)$ of MinVC, and let $G_C = (V_C, E_C)$ be the corresponding instance of MCC. G_C is a rooted tree, defined as follows. First, we define $|V|$ paths, one for each vertex in G. Path P_i contains vertex $v_{c,i}$, colored by c_i, and vertices $e_{c,i,j}$, for each $\{v_i, v_j\} \in E$, colored by c_{ij}. Notice that vertices

$e_{c,i,j}$ appears in P_i based on the lexicographic order of the corresponding edges. Moreover, there exist two vertices associated with edge $\{v_i, v_j\} \in E$, namely $e_{c,i,j}$ (in P_i) and $e_{c,j,i}$ (in P_j), which are both colored by c_{ij}. The tree G_C is obtained by connecting the paths $P_1, \ldots P_{|V|}$ to a root r, which is colored by c_r, where c_r is a fresh new color (see Figure 2).

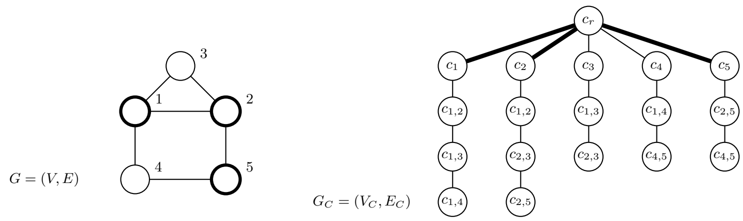

Fig. 2. Sample construction of an instance of MCC from an instance of MinVC. A possible solution for MinVC is given in thick while edges to be cut for the instance of MCC are also in thick.

Lemma 8. *Let $G = (V, E)$ be an instance of MinVC, and let $G_C = (V_C, E_C)$ be the corresponding instance of MCC. Then: (1) given a vertex cover of G of size k, we can compute in polynomial time a solution of MCC over instance G_C consisting of $k+1$ colorful components; (2) given a solution of MCC over instance G_C consisting of $k+1$ colorful components, we can compute in polynomial time a vertex cover of G of size k.*

By the previous lemma, the following result holds.

Theorem 9. *MCC is NP-hard even when the input graph is at distance* 1 *to Disjoint Paths.*

It is worth noticing that this result extends to parameter pathwidth or distance to interval graph, as these last parameters are "stronger" than distance to disjoint path in the sense of [11].

5 An FPT Algorithm for MEC Parameterized by k

We present a parameterized algorithm for MEC with respect to the natural parameter k. To do so, we will show that the problem admits an exponential size kernel.

Given a colored graph G, we first compute a Depth-First-Search (DFS) $D = (V, E_D, E_B)$ of G. Recall that D consists of a tree induced by $D' = (V, E_D)$ (hence not considering edges in E_B), while $E_B = E \setminus E_D$ are called *backward edges* and have the following well-known property (see [6] for details).

Lemma 10. *Consider a graph G and the corresponding DFS $D = (V, E_D, E_B)$. Let $\{u, v\}$ be a backward edge. Then there exists a path p in $D' = (V, E_D)$ that starts in the root of D' and contains both u and v.*

We will first show some easy cases where there is a solution of MEC of size at least k. Let V_A be the set of vertices of V which are parent of a leaf in D'.

Lemma 11. *If there exists a path in D' from the root $r(D')$ to a leaf of D' of length at least $2k$, then there exists a solution of MEC of size at least k.*

Lemma 12. *There exists a solution of MEC of size at least k if $|V_A| \geqslant k$.*

Now, for each vertex $v \in V_A$ we consider the leaves adjacent to v and their colors. Define the set $C_x(v)$ as the set of leaves colored by c_x and adjacent to $v \in V_A$ in D. Then the following property holds.

Lemma 13. *Given a vertex $v \in V_A$, if there exist $\sqrt{2k}$ non-empty sets $C_x(v)$ associated with distinct colors c_x, then there exists a solution of MEC of size at least k.*

Given a vertex $v \in V_A$ and a set $C_x(v)$, consider the sets of vertices connected with backward edges to a vertex $u \in C_x(v)$. Define $Adj(C_x(v)) = \{V'_A \subseteq V_A : \{u, w\} \in E, u \in V'_A, w \in C_x(v)\}$.

The following property holds.

Lemma 14. *Given a vertex-colored graph G such that the hypothesis of Lemma 11 does not hold, consider a vertex v in V_A and a set $C_x(v)$. of possible sets of adjacent vertices to a node u of $C_x(v)$.*

Based on Lemma 14, we can partition the vertices of each $C_x(v)$ into sets $C_{x,1}(v), \ldots, C_{x,p}(v)$, with $p \leqslant 2^{2k+1}$, depending on their set of adjacent vertices (that is two vertices of $C_x(v)$ belong to the same set $C_{x,t}(v)$ if they have the same set of adjacent vertices).

Now, assume that the hypotheses of Lemma 11, Lemma 12 and Lemma 13 do not hold. Consider an algorithm that, for each set $C_{x,i}(v)$, computes a set $C'_{x,i}(v)$ by picking at most k vertices of $C_{x,i}(v)$ and removing the other vertices of $C_{x,i}(v)$. Let G' be the resulting graph. We claim that G' contains at most $O(k^2 2^{2k+1})$ vertices. First, notice that each $C'_{x,t}(v)$ contains at most k vertices and that, for each vertex v, there exists at most 2^{2k+1} sets $C'_{x,t}(v)$. Since, there exist at most $O(k\sqrt{k})$ sets $C_x(v)$ (at most $\sqrt{2k}$ colors c_x and at most k vertices $v \in V_A$), we can conclude that G' contains at most $O(k^2\sqrt{k}2^{2k+1})$ vertices in sets $C'_{x,i}(v)$.

Now, consider the vertices G' which are not contained in some set $C'_{x,i}(v)$. These vertices correspond to internal vertices of D'. Since the hypothesis of Lemma 11 does not hold, D' is a tree of depth at most $2k$, and there exist at most k vertices adjacent to leaves, as $|V_A| < k$. Hence there exist at most k paths of length $2k$ in D' from the root to vertices adjacent to leaves, thus we can conclude that there exist at most $2k^2$ internal vertices in D'. Hence there exists at most $2k^2$ vertices in G' which are not contained in some set $C'_{x,i}(v)$.

Now, we prove that (G', k) is a kernel for MEC.

Lemma 15. *There exists a collection of disjoint colorful components $V_1, \ldots, V_h$ of size at least 2 in G if and only if there exists a collection of disjoint colorful components $V'_1, \ldots, V'_h$ in G', with $|V_i| = |V'_i|$, $1 \leqslant i \leqslant h$.*

Hence we have the following result.

Theorem 16. *There exists a kernel of size $O(k^2\sqrt{k}2^{2k+1})$ for MEC.*

6 A poly-kernel for MEC on trees

In this section, we show that in the special case of MEC where the input graph is a tree, the kernel size can be quadratic. The algorithm is similar to the one of Section 5. Consider a colored tree $G_T = (V, E, C)$, and let $r(G_T)$ denote the root of G_T. Lemmata 11,12,13 hold for G_T. Hence, we focus only on the leaves of G_T.

Since G_T is a tree, it follows that a leaf u having ancestor v belongs to a component of size at least 2 only if u and v belongs to the same component. It follows that among the leaves having color c_x and adjacent to a vertex u, only one can belong to a colorful component of size at least 2. Hence, given $v \in V_A$, let $C_x(v)$ be the set of leaves adjacent to v and colored by c_x. We remove all but one vertex from $C_x(v)$. Let G'_T be the resulting tree. We have the following property for G'_T.

Lemma 17. *There exists a collection of disjoint colorful components $V_1, \ldots, V_h$ of size at least 2 in G_T if and only if there exists a collection of disjoint colorful components $V'_1, \ldots, V'_h$ in G'_T, with $|V_i| = |V'_i|$, $1 \leqslant i \leqslant h$.*

Theorem 18. *There exists a kernel of size $O(k^2)$ for MEC on trees.* $k^2\sqrt{k}$?

Proof. The result follows from Lemma 17 and from the fact that tree G'_T contains at most k^2 internal vertices (by Lemma 11 and by Lemma 12) and there exist at most $O(k\sqrt{k})$ sets $C_{x,i}(v)$ (by Lemma 13), each of size bounded by 1. □

7 Structural parameterization of MEC

It is easy to see that the results on structural parameterization for MCC hold also for MEC (after appropriate modifications).

Theorem 19. *MEC on paths can be solved in $O(n^3)$-time.*

Similarly to MCC, MEC is NP-hard even if we restrict the instance to graphs having distance 1 to Disjoint Paths. As for MCC, it is worth noticing that this hardness result extends to other stronger parameters like pathwidth [11].

Theorem 20. *MEC is NP-hard even when the input graph has distance 1 to Disjoint Paths.*

8 Conclusion

In the future, we aim at refining the parameterized complexity analysis, for example deepen the structural results for MCC and MEC. Moreover, it would be interesting to study the parameterized complexity of the two problems under other meaningful parameters in the direction of parameterizing above a guaranteed value.

References

1. Adamaszek, A., Blin, G., Popa, A.: Approximation and hardness results for the maximum edges in transitive closure problem. In: Kratochvíl, J., Miller, M., Froncek, D. (eds.) Combinatorial Algorithms - 25th International Workshop, IWOCA 2014. LNCS, vol. 8986, pp. 13–23. Springer (2014)
2. Adamaszek, A., Popa, A.: Algorithmic and hardness results for the colorful components problems. Algorithmica 73(2), 371–388 (2015)
3. Ausiello, G., Crescenzi, P., Gambosi, G. Kann, V., Marchetti-Spaccamela, A., Protasi, M.: Complexity and Approximation: Combinatorial optimization problems and their approximability properties. Springer-Verlag (1999)
4. Betzler, N., van Bevern, R., Fellows, M.R., Komusiewicz, C., Niedermeier, R.: Parameterized algorithmics for finding connected motifs in biological networks. IEEE/ACM Trans. Comput. Biology Bioinform. 8(5), 1296–1308 (2011)
5. Bruckner, S., Hüffner, F., Komusiewicz, C., Niedermeier, R., Thiel, S., Uhlmann, J.: Partitioning into colorful components by minimum edge deletions. In: Kärkkäinen, J., Stoye, J. (eds.) Combinatorial Pattern Matching - 23rd Annual Symposium, CPM 2012. LNCS, vol. 7354, pp. 56–69. Springer (2012)
6. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms (3. ed.). MIT Press (2009)
7. Dondi, R., Fertin, G., Vialette, S.: Complexity issues in vertex-colored graph pattern matching. J. Discrete Algorithms 9(1), 82–99 (2011)
8. Dondi, R., Fertin, G., Vialette, S.: Finding approximate and constrained motifs in graphs. Theor. Comput. Sci. 483, 10–21 (2013)
9. Downey, R.G., Fellows, M.R.: Fundamentals of Parameterized Complexity. Springer-Verlag (2013)
10. Garg, N., Vazirani, V.V., Yannakakis, M.: Primal-dual approximation algorithms for integral flow and multicut in trees. Algorithmica 18(1), 3–20 (1997)
11. Komusiewicz, C., Niedermeier, R.: New races in parameterized algorithmics. In: Rovan, B., Sassone, V., Widmayer, P. (eds.) Mathematical Foundations of Computer Science 2012 - 37th International Symposium, MFCS 2012, Bratislava, Slovakia, August 27-31, 2012. Proceedings. LNCS, vol. 7464, pp. 19–30. Springer (2012)
12. Lacroix, V., Fernandes, C.G., Sagot, M.: Motif search in graphs: Application to metabolic networks. IEEE/ACM Trans. Comput. Biology Bioinform. 3(4), 360–368 (2006)
13. Zheng, C., Swenson, K.M., Lyons, E., Sankoff, D.: OMG! Orthologs in Multiple Genomes - Competing Graph-Theoretical Formulations. In: Przytycka, T.M., Sagot, M. (eds.) Algorithms in Bioinformatics - 11th International Workshop, WABI 2011. LNCS, vol. 6833, pp. 364–375. Springer (2011)