Enumeration of Minimal Tropical Connected Sets

Ivan Bliznets[1(✉)], Danil Sagunov[2], and Eugene Tagin[3]

[1] Utrecht University, Utrecht, Netherlands
iabliznets@gmail.com,i.bliznets@uu.nl

[2] St. Petersburg Department of Steklov Institute of Mathematics of the RAS, Saint Petersburg, Russia

[3] St. Petersburg State University, Saint Petersburg, Russia

Abstract. A subset of vertices in a vertex-colored graph is called tropical if vertices of each color present in the subset. This paper is dedicated to the enumeration of all minimal tropical connected sets in various classes of graphs. We show that all minimal tropical connected sets can be enumerated in $\mathcal{O}(1.7142^n)$ time on n-vertex interval graph which improves previous $\mathcal{O}(1.8613^n)$ upper bound obtained by Kratsch et al. Moreover, for chordal and general class of graphs we present algorithms with running times in $\mathcal{O}(1.937^n)$ and $\mathcal{O}(1.999958^n)$, respectively. The last two algorithms answer question implicitly asked in the paper [Kratsch et al. SOFSEM 2017]: «Is the number of tropical sets significantly smaller than the trivial upper bound 2^n?».

Keywords: tropical sets · enumeration algorithms · graph motif · chordal graphs · beating brute-force

1 Introduction

Efficient enumeration of objects with special properties is an important problem in computer science. There are many problems in graph theory in which the answer is a list of subsets of vertices that have a certain property or the cardinality of this set. Most often one is looking for the inclusion minimal/maximal induced subgraphs with additional attributes. The most classical result is that all maximal independent sets can be enumerated in $\mathcal{O}^*(3^{\frac{n}{3}})$ time [32], moreover, the running time is tight since there are graphs that have $3^{\frac{n}{3}}$ maximal independent sets [32]. It is also known that all minimal dominating sets can be listed in $\mathcal{O}(1.7159^n)$ time [21]. If the input graph is restricted to a special type of graphs like trees, chordal, and interval graphs faster algorithms were designed. For example, in chordal graphs, all minimal dominating sets can be enumerated

Work of Ivan Bliznets is supported by the project CRACKNP that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 853234).

M. Mavronicolas (Ed.): CIAC 2023, LNCS 13898, pp. 127–141, 2023.
https://doi.org/10.1007/978-3-031-30448-4_10

in $\mathcal{O}(1.5048^n)$ time [24], in trees in $\mathcal{O}(1.4656^n)$ time [28] and in interval graphs in $\mathcal{O}^*(3^{\frac{n}{3}})$ time [11] ($\mathcal{O}^*()$ suppress polynomial factors in the same way as $\mathcal{O}$ suppress constant factors).

Listing all potential candidates might be essential in some applications and the enumeration algorithms perform exactly this task. Such algorithms sometimes are key ingredients of very efficient algorithms for certain problems. For example, Lawler's $\mathcal{O}^*((1+3^{\frac{1}{3}})^n)$ algorithm [30] for a chromatic number is based on the fact that all maximal independent sets can be enumerated within $\mathcal{O}^*(3^{\frac{n}{3}})$ running time. The same fact is also used in the fastest known algorithm for 4-coloring by Fomin, Gaspers, and Saurabh [18]. Construction of efficient enumeration algorithms also often leads to new combinatorial upper bounds on the number of objects with special properties.

In the paper, we consider a problem of enumeration of subsets in vertex-colored graphs. More precisely, we are interested in enumeration of tropical subsets with additional properties. A set of vertices is called tropical if vertices of each color are presented in the set. There are papers dedicated to the study of various variants of tropical sets. For example tropical dominating sets were studied in [16], tropical matchings in [10], tropical paths in [9], tropical vertex-disjoint cycles in [31]. However, it seems that tropical connected sets attract the greatest attention [7,8,17,27]. Most probably that can be explained by close connection of connected tropical sets with a Graph Motif problem that was motivated by applications in biological network analysis [29] and later found applications in social networks [2] and in the context of mass spectrometry [5].

Angles d'Auriac et al. [17] proved that finding a minimum tropical connected set is NP-complete even on trees of height three as well as on split and interval graphs. An exact exponential-time algorithm for Minimum Tropical Connected Set was presented by Chapelle et al. [8]. In the case of a general input graph, they provide a $\mathcal{O}(1.5359^n)$ algorithm while in the case of trees they give a $\mathcal{O}(1.2721^n)$ algorithm. Later focus was shifted to enumeration of all minimal tropical connected sets and Kratsch et al. [27] presented algorithms tailored to special types of input graphs. So for split graphs they constructed a $\mathcal{O}(1.6402^n)$ algorithm, for interval graphs $\mathcal{O}(1.8613^n)$ time algorithm, for co-bipartite graphs and block graphs a $\mathcal{O}^*(3^{\frac{n}{3}})$ algorithm was presented. Moreover, in the same paper, several lower bounds on the maximum number of minimal tropical connected sets were given: for co-bipartite, interval, and block graphs the lower bound was $3^{\frac{n}{3}}$, for split graphs it was 1.4766^n and for chordal graphs it was 1.4916^n. No algorithm was presented for the case of a general input graph or for the case when the input graph is known to be chordal. We present a quote from the paper by Kratsch et al. [27]: "Interestingly, the best known upper bound for the maximum number of minimal tropical connected sets in an arbitrary graph and even for chordal graphs is the trivial one which is 2^n". The main goal of our paper is to answer this implicit question and present the first non-trivial upper bounds on the maximum number of minimal connected tropical sets in chordal and general graphs. We note that these types of questions, i.e. whether there is an algorithm faster than naive brute-force search play a tremendous role in computer science, especially in areas like fine-grained complexity, parameterized algorithm, and exact exponential algorithms. For many problems, it is easy to come up with algorithms significantly faster than brute-force search. However, for some problems, even a tiny improvement over brute-force search is a highly non-trivial task [1,3,4,13–15,19,20,33]. Moreover, there are a lot of important problems for which we do not know algorithms faster than simple brute-force search. For example, Set Cover problem, Satisfiability, and Orthogonal Vectors. Moreover, it is conjectured that it is impossible to construct such algorithms: Orthogonal Vector Conjecture [35], Set Cover Conjecture [12], Strong Exponential Time Hypothesis [12].

As a result of our research, we present an algorithm that enumerates all minimal tropical connected sets in $\mathcal{O}(1.999958^n)$ time in general graphs and an algorithm that performs the same task on chordal graphs in $\mathcal{O}(1.937^n)$ time. Moreover, we present an algorithm for interval graphs that runs in $\mathcal{O}(1.7142^n)$ time, which improves the previous asymptotic upper bounds of $\mathcal{O}(1.8613^n)$.

2 Preliminaries

We consider finite undirected graphs without loops or multiple edges. For graph G, $V(G)$ is the set of vertices of G, E(G) is a set of edges of G and $n = |V(G)|$ if not stated otherwise. $N(v)$ is the set of neighbours of vertex $v \in V(G)$. $N[v] = N(v) \cup \{v\}$ is the set of neighbours of vertex v including itself. For a set of vertices $X \subseteq V(G)$, $N_G[X] = \cup_{v \in X} N_G[v]$ and $N_G(X) = N_G[X] \setminus X$. For a subset $X \subseteq V(G)$ of vertices, $G[X]$ denotes the subgraph of G induced by X. A clique is a subset of vertices $D \subseteq V(G)$ such that $G[D]$ is a complete graph. *Chordal graph* is a graph without induced cycles of length bigger than 3. Chordal graphs admit many equivalent definitions, more details can be found here [26]. *Interval graphs* is a subclass of chordal graphs in which each vertex can be assigned an interval on a line such that two vertices have a common edge if and only if the corresponding intervals overlap [6]. $c : V(G) \to \mathbb{N}$ is a coloring function (not necessary proper), which assigns to each vertex a certain color. Let $c(X) = \{c(v) : v \in X\}$ be a set of different colors assigned to vertices of $X \subseteq V(G)$. Let $\mathcal{C} = c(V(G))$ be a set of all colors of graph G. We assume that $\mathcal{C} = \{1, 2, \ldots, C\}$. *A tropical set* of graph G is a subset of vertices $X \subseteq V(G)$ such that $c(X) = c(V(G))$. *A tropical connected set* of graph G is a subset of vertices $X \subseteq V(G)$ such that X is tropical and $G[X]$ is a connected subgraph. Let $\gamma = \frac{|\mathcal{C}|}{n}$. *A rainbow set* is a tropical set of the smallest size $|\mathcal{C}|$, i.e. a set that contains each color exactly once. A subset of vertices $X \subseteq V(G)$ is called *minimal tropical connected set* if there is no $Y \subsetneq X$ such that Y is tropical and $G[Y]$ is a connected subgraph.

Let $n, \ell, C, n_1, n_2, \ldots n_C$ be positive integers such that $n_1 + n_2 + \cdots + n_C = n$. We denote by $P^{n_1,n_2,\ldots,n_C}_{n,\ell,C}$ the number of tuples $(a_1, a_2, \ldots, a_C) \in \mathbb{Z}^C_{>0}$ such that $a_1 + a_2 + \cdots + a_C = \ell$ and $1 \leq a_i \leq n_i$ for each $1 \leq i \leq C$. Let $P_{n,\ell,C} = \max_{n_1,\ldots,n_C} P^{n_1,n_2,\ldots,n_C}_{n,\ell,C}$.

We assume that G is connected. Since otherwise, we can simply run our algorithms on each connected component of G separately and output a union of the obtained results. In our algorithms we use upper bounds on the number of tropical, rainbow sets and binomial coefficients given in the lemmas below. Due to the space constraints, proofs of lemmas marked by ($\star$) are omitted.

Lemma 1. [22] *For any positive integer n and $0 \leq \alpha \leq 1$ we have $\binom{n}{\alpha n} \leq 2^{H(\alpha)n}$, where $H(\cdot)$ is the binary entropy function i.e. $H(x) = -x\log_2(x) - (1-x)\log_2(1-x)$.*

Lemma 2 ($\star$). *Let G be a colored graph with n vertices and a number of used colors is γn then:*

1. *the number of all rainbow sets is at most $(\frac{1}{\gamma})^{\gamma n}$;*
2. *the number of tropical sets is at most $(2^{\frac{1}{\gamma}} - 1)^{\gamma n}$.*

Moreover, all rainbow and tropical sets can be listed almost within the same running time i.e. within $\mathcal{O}^*((\frac{1}{\gamma})^{\gamma n})$ and $\mathcal{O}^*((2^{\frac{1}{\gamma}} - 1)^{\gamma n})$ running time.

Lemma 3 ($\star$). *If $n_1, n_2, \ldots, n_k$ are positive integer numbers such that $n_1 + n_2 + \cdots + n_k = n$ then $n_1 n_2 \ldots n_k \leq 3^{\frac{n}{3}}$.*

Lemma 4 ($\star$). *For any positive integers n, ℓ, C we have: (i) $P_{n,\ell,C} \leq (\frac{n}{C})^C$; (ii) $P_{n,\ell,C} \leq \binom{\ell-1}{C-1}$.*

Lemma 5 ($\star$). *Let (G, c) be a colored graph and $S \subseteq V(G)$. There is a polynomial time algorithm that tests whether S is a* Minimal Tropical Connected Set.

3 General Graphs

In this section we present an algorithm that enumerates all inclusion-minimal tropical connected sets. The running time of the algorithm is $\mathcal{O}(1.999958^n)$. Hence, the number of Minimal Tropical Connected Sets is at most $\mathcal{O}(1.999958^n)$. These results answers an implicit question from [27], where trivial upper bound 2^n was given. In order to present the algorithm with mentioned running time we construct two auxiliary algorithms for the problem. The first one is given in the lemma below.

Lemma 6. *Let G be a vertex-colored graph with n vertices colored with $C = \gamma n$ colors. There is an algorithm that enumerates all* Minimal Tropical Connected Sets *in $\mathcal{O}^*((2^{1/\gamma} - 1)^{\gamma n})$ time.*

Proof. From Lemma 2, it follows that the number of all tropical sets is at most $((2^{1/\gamma} - 1)^{\gamma n})$. It is straightforward to enumerate all of them within this running time. What is left is to delete all sets that are not minimal tropical connected. However, by Lemma 5 we can run such test for each candidate in polynomial time. Hence in $\mathcal{O}^*((2^{1/\gamma} - 1)^{\gamma n})$ time we can list all Minimal Tropical Connected Sets. □

Before we proceed to the second auxiliary algorithm we state the following definition and theorem from [34].

Definition 1. [34] *For a given subset of vertices T we call a superset S of T* T-connecting *if S induces a connected graph. Moreover, we call S a minimal T-connecting if no strict subset of S is T-connecting.*

Theorem 1. [34] *For an n vertex graph $G = (V, E)$ and a terminal set $T \subseteq V$ where $|T| \leq \frac{n}{3}$ there are at most $\binom{n-|T|}{|T|-2} \cdot 3^{(n-|T|)/3}$ minimal T-connecting vertex sets and they can be enumerated in time $\mathcal{O}^*(\binom{n-|T|}{|T|-2} \cdot 3^{(n-|T|)/3})$.*

Lemma 7 (⋆). *For an n vertex graph $G = (V, E)$ and a terminal set $T \subseteq V$ there are at most $2^{n-|T|}$ minimal T-connecting vertex sets and they can be enumerated in time $\mathcal{O}^*(2^{n-|T|})$.*

Equipped with the previous theorem and lemma, we are ready to prove the following result.

Lemma 8. *Let (G, c) be a graph with n vertices colored in $C = \gamma n$ colors and $\gamma \leq \frac{1}{3}$. There is an algorithm that enumerates all* Minimal Tropical Connected Sets *in time*

$$\max\{\max_{\alpha:\gamma\leq\alpha\leq 1-2\gamma} 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n} \cdot \min\{2^{H(\frac{\gamma}{1-\alpha})\cdot(1-\alpha)n} \cdot 3^{\frac{1-\alpha}{3}n}, 2^{(1-\alpha)n}\}, \max_{\alpha:1-2\gamma\leq\alpha\leq 1} 2^{H(\frac{\gamma}{\alpha})} \cdot 2^{(1-\alpha)n}\}$$

up to a polynomial factor.

Proof. Recall that our graph contains vertices of $C = \gamma n$ different colors and the number of vertices colored in the i-th color is exactly n_i, i.e. $n_1 + n_2 + \cdots + n_C = n$. Let $V_i = \{v_i^1, \ldots, v_i^{n_i}\}$ be a set of all vertices of the i-th color.

We know that any tropical set must contain a rainbow set. With each minimal tropical connected set X we associate a rainbow set R_X constructed in the following way: for each $i \in \{1, 2, 3, \ldots, C\}$ we put v_i^j in R_X if $v_i^j \in X$ and for each $p < j$ we have that $v_i^p \notin X$. We note that $X \setminus R_X$ is an inclusion-minimal set that connects vertices from R_X, otherwise X is not minimal tropical connected set.

Now we are ready to describe the algorithm. In the first step we list all potential candidates for the role of R_X. So, basically, we consider many branchings and each branch defines a corresponding R_X. So in branch with $R_X = \{v_1^{j_1}, v_2^{j_2}, \ldots, v_C^{j_C}\}$ we assume that R_X is part of a minimal tropical connected set, while vertices $v_i^{p_i}$ with $p_i < j_i$ are not, hence in this branch these vertices can be simply deleted from the graph. At this point at each branch we already decided about $\ell = j_1 + j_2 + \cdots + j_C$ vertices whether they belong to a minimal tropical connected set or not. There are $n - j_1 - j_2 - \cdots - j_C$ vertices that are left, let us call the set of these vertices W. Now it is enough to list all inclusion-minimal sets $Y' \subseteq W$ such that $R_X \cup Y'$ is connected and discard those sets that are not minimal tropical connected sets. Check whether a set is a minimal tropical connected set can be done in a polynomial time by Lemma 5. Moreover, by Theorem 1 and Lemma 7 we can list all Y' that connect R_X in time $\min\{\mathcal{O}^*(\binom{|W|}{|C|-2} \cdot 3^{\frac{|W|}{3}}), \mathcal{O}^*(2^{|W|})\}$ if $|W| \geq 2C$ or in time $\mathcal{O}^*(2^{|W|})$ otherwise. Denote by

$$f(w,c) = \begin{cases} \min\left(\binom{w}{c-2} \cdot 3^{\frac{w}{3}}, 2^w\right), & \text{if } w \geq 2c \\ 2^w, & \text{otherwise} \end{cases}$$

So the running time of the algorithm up to a polynomial factor is equal to:

$$\sum_{\substack{1 \leq j_1 \leq n_1 \\ \cdots \\ 1 \leq j_C \leq n_C}} f(n - (j_1 + j_2 + \cdots + j_C), C).$$

Recall that $P_{n,\ell,C} = \max_{n_1,n_2,\ldots,n_C} P^{n_1,n_2,\ldots,n_C}_{n,\ell,C}$ and $P^{n_1,n_2,\ldots,n_C}_{n,\ell,C}$ is the number of tuples $(a_1,\ldots,a_C)$ such that $a_1 + a_2 + \cdots + a_C = \ell$ and $1 \leq a_i \leq n_i$. So, the running time can be rewritten (up to a polynomial factor) as $\sum_{C \leq \ell \leq n} P_{\ell,C} \cdot f(n-\ell, C)$. By Lemma 4 we know that $P_{n,\ell,C} \leq \binom{\ell-1}{C-1} \leq \binom{\ell}{C}$

So, the running time up to the polynomial factor is bounded by $\max_{C \leq \ell \leq n} \binom{\ell}{C} \cdot f(n-\ell, C)$. Since $\gamma \leq \frac{1}{3}$ we know that $C \leq n - 2C$. So we can split interval $[C, n]$ into two intervals $[C, n-2C]$ and $[n-2C, n]$. So, it is obvious that:

$$\max_{C \leq \ell \leq n} \binom{\ell}{C} \cdot f(n-\ell, C) =$$

$$\max\left\{\max_{C \leq \ell \leq n-2C} \binom{\ell}{C} \cdot f(n-\ell, C), \max_{n-2C \leq \ell \leq n} \binom{\ell}{C} \cdot f(n-\ell, C)\right\} =$$

$$\max\left\{\max_{C \leq \ell \leq n-2C} \binom{\ell}{C} \cdot \min\left\{\binom{n-\ell}{C-2} \cdot 3^{\frac{n-\ell}{3}}, 2^{n-\ell}\right\}, \max_{n-2C \leq \ell \leq n} \binom{\ell}{C} \cdot 2^{n-\ell}\right\}.$$

Let $\ell = \alpha n$, recall that $C = \gamma n$. Note that $\binom{w}{c-2} \leq w^2 \binom{w}{c}$ for any w, c and $\binom{n}{\beta n} \leq 2^{H(\beta)n}$ for arbitrary $0 \leq \beta \leq 1$. Keeping the above said in mind, the running time up to the polynomial factor is bounded by:

$$\max\{\max_{\alpha:\gamma \leq \alpha \leq 1-2\gamma} 2^{H(\frac{\gamma}{\alpha})\cdot \alpha n} \cdot \min\{2^{H(\frac{\gamma}{1-\alpha})\cdot(1-\alpha)n} \cdot 3^{\frac{1-\alpha}{3}n}, 2^{(1-\alpha)n}\},$$
$$\max_{\alpha:1-2\gamma \leq \alpha \leq 1} 2^{H(\frac{\gamma}{\alpha})} \cdot 2^{(1-\alpha)n}\}$$

So, we obtain the desired result. □

Now, we have all tools to show the main result of this section.

Theorem 2. *Let G be a colored graph with n vertices. There is an algorithm that enumerates all* Minimal Tropical Connected Sets *in time $\mathcal{O}(1.999958^n)$. Hence, the number of all* Minimal Tropical Connected Sets *in a graph on n vertices is at most $\mathcal{O}(1.999958^n)$.*

Proof. In order to construct an algorithm with desired running time, we carefully choose the right algorithm from algorithms presented in Lemmas 6 and 8. Note that $(2^{1/\gamma} - 1)^{\gamma}$ is decreasing function for $\gamma \in [0, 1]$, see Fig. 2. So it is more reasonable to use the algorithm from Lemma 6 when γ is large enough, i.e. input graph G has a sufficiently large number of colors. In contrary, if we plot the function

$$\max\{\max_{\alpha:\gamma\leq\alpha\leq1-2\gamma} 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n} \cdot \min\{2^{H(\frac{\gamma}{1-\alpha})\cdot(1-\alpha)n} \cdot 3^{\frac{1-\alpha}{3}n}, 2^{(1-\alpha)n}\},$$
$$\max_{\alpha:1-2\gamma\leq\alpha\leq1} 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n} \cdot 2^{(1-\alpha)n}\},$$

we see that the function is non-decreasing for $\gamma \in [0, 0.1]$, Fig. 2, so the second algorithm shows its best performance when the number of different colors is small.

So if the number of different colors in graph G is bigger than $0.08369n$ we run the first algorithm, i.e. if $\gamma \geq 0.08369$ then we run the algorithm with running time $\mathcal{O}^*((2^{1/\gamma} - 1)^{\gamma n}) \leq \mathcal{O}(1.999958^n)$. Otherwise, we run the second algorithm with running time

$$\max\{\max_{\alpha:\gamma\leq\alpha\leq1-2\gamma} 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n} \cdot \min\{2^{H(\frac{\gamma}{1-\alpha})\cdot(1-\alpha)n} \cdot 3^{\frac{1-\alpha}{3}n}, 2^{(1-\alpha)n}\},$$
$$\max_{\alpha:1-2\gamma\leq\alpha\leq1} 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n} \cdot 2^{(1-\alpha)n}\} \leq \mathcal{O}^*(1.999958^n).$$

So, in any case we get the desired running time. □

4 Chordal Graphs

The objective of this section is to present an algorithm that enumerates all Minimal Tropical Connected Sets in chordal graphs within $\mathcal{O}(1.937^n)$ running time which is smaller than in the case of arbitrary graphs. As a consequence we get that the number of all Minimal Tropical Connected Sets in any colored chordal graph is at most $\mathcal{O}(1.937^n)$. We note that this answers an implicit question from [27] where even for chordal graphs, the trivial 2^n bound was the only given bound on the number of minimal tropical connected sets. In order to achieve this improvement compared to the case of general input graph we replace algorithm described in Lemma 2 with a more efficient one. Instead of enumerating T-connecting sets we will be interested in enumerating connected dominating sets in special chordal subgraphs.

Before we proceed we recall some properties of chordal graphs and tree-decomposition.

A tree decomposition of a graph G is a pair $(\{X_i \mid i \in I\}, T = (I, F))$ with $\{X_i \mid i \in I\}$ a collection of subsets of $V(G)$, called *bags*, and $T = (I, F)$ a tree, such that

1. For every $v \in V(G)$, there exists $i \in I$ with $v \in X_i$.

2. For every $\{v, w\} \in E$, there exists $i \in I$ with $v, w \in X_i$
3. For every $i, j, k \in I$, if j is contained in a path from i to k in T, then $X_i \cap X_k \subseteq X_j$.

The following lemma is folklore and easily follows from lemma 7.1 in [12].

Lemma 9. [12] *Let $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ be a tree decomposition of non-complete graph G and let u, v, w be nodes in tree T with bags B_u, B_v, B_w such that shortest path from u to w in tree T goes through vertex v. If $x \in B_u, y \in B_w$ then there is no path from x to y in graph $G \setminus B_v$ (note the statement trivially holds if x or y belongs to B_v).*

The following lemma is well known [25].

Lemma 10. [25] *Let G be a chordal graph then there exists a tree decomposition of G in which all bags are cliques. Moreover, such decomposition can be constructed in polynomial time.*

Now we are ready to present relevant results about connected dominating sets.

Definition 2. *For a connected graph G a subset of vertices $X \subseteq V(G)$ is called* connected dominating set*, if X induces a connected subgraph and $N[X] = V(G)$.*

Theorem 3. [23] *Any chordal graph with n vertices has no more than 1.4736^n minimal connected dominating sets. And all of them can be enumerated within $\mathcal{O}(1.4736^n)$ running time.*

Before we proceed with the algorithm we prove several auxiliary lemmas.

Lemma 11. *Let X be a vertex subset in graph G. Let S be a minimal set that connects X, i.e. is an inclusion-minimal subset of $V(G)$ such that the induced subgraph $G[S \cup X]$ is connected. If $S \cup X$ is a dominating set then there is a minimal connected dominating subset $M \subseteq V(G)$ such that $S \subseteq M \subseteq S \cup X$.*

Proof. We know that $S \cup X$ is connected and a dominating set. So it must contain some minimal connected dominating set. Let us call this set M'. If $S \subseteq M'$ then we are done and can take $M = M'$. If this is not the case, consider $S' = M' \setminus X$. Since $M' \subseteq (S \cup X)$ and $S \not\subseteq M'$ we have that $S' \subsetneq S$. M' is connected and dominating so $M' \cup X$ is also connected. Since $S' \cup X = M' \cup X$ we have that S is not an inclusion-minimal subset of $V(G)$ such that $G[S \cup X]$ is connected. So we get a contradiction. Hence, S must be a subset of M'. And we can take $M = M'$. □

Definition 3. *Let G be a chordal graph. For a subset of vertices $X \subseteq V(G)$ we call an X-*restriction *a chordal graph G_X obtained in the following way:*

1. *Take the tree decomposition $\mathcal{T}$ of G where each bag is a clique, one that is described in Lemma 10*

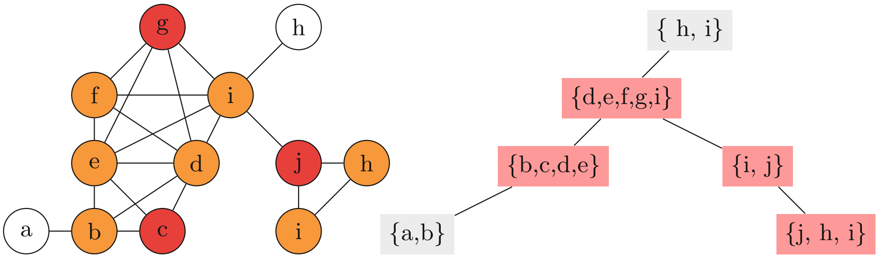

Fig. 1. The left part of figure shows graph G, set $X = \{c, g, j\}$ colored red and subgraph G_X that consist of orange and red vertices. The right part of figure shows tree-decomposition T of graph G, and an inclusion-minimal subtree T_X with all bags containing vertices from X.

2. Find inclusion-minimal subtree $T_X \subseteq T$ such that T_X includes all nodes whose bags contain vertices from X
3. G_X is a graph, obtained by removing all vertices in G that are not contained in bags of T_X.

Note that G_X is a chordal graph (as an induced subgraph of a chordal graph) and T_X is a tree-decomposition of G_X. Illustration of this definition is presented in Fig. 1.

Lemma 12. *Let G be a chordal graph and $X \subseteq V(G)$, then all minimal connected subgraphs of G containing X must be subgraphs of a restriction G_X.*

Proof. Let $\mathcal{T}$ be the tree decomposition of G (with corresponding tree T) constructed by the algorithm from Lemma 10 and T_X be a subtree in the decomposition that we used to construct an X-restriction graph G_X. Assume that there is a connected minimal subgraph H such that $X \subseteq V(H)$ and $V(H) \nsubseteq V(G_X)$. It can happen only if there are $x, y \in X$ such that there exists an vertex inclusion-minimal path p in $V(H)$ that connects x, y and has vertices outside G_X. Let us consider the shortest such path p and let z be a vertex on it that does not belong to G_X. Consider the shortest path $p_T = v_1, v_2, \ldots, v_q$ in T from the subtree T_X to the subtree T_z induced by bags containing z. Since, $z \notin G_X$ we have $T_X \cap T_z = \emptyset$. Hence, there is only one such path as otherwise T is not a tree. We note that $v_1 \in T_X$. Consider a subset of vertices S' from $V(G)$ that forms a bag of vertices for node v_1. S' is a separator and any path going from a vertex in G_X to z must pass through one of the vertices in S'. So it means that path $p = x, \ldots, z, \ldots y$ must contain vertices $u_1, u_2 \in S'$ on the subpaths from x to z and from z to y. However, $u_1, u_2 \in S'$ and S' is a clique as a bag of a node in a tree-decomposition for a chordal graph. So it means that the path p can be shorten as instead of going from u_1 to z and from z to u_2 we can go straight-ahead from u_1 to u_2. This leads to a desired contradiction. □

Lemma 13. *Let G be a chordal graph, $X \subseteq V(G)$, G_X be an X-restriction of G. If Y is connected in G_X and $X \subseteq Y$ then Y is a dominating set in G_X.*

Proof. It is enough to show that each bag of tree T_X contains a vertex from Y. Indeed, any vertex $v \in V(G_X)$ belongs to some bag B in the tree T_X. Since all bags in T_X are cliques, v is dominated by any vertex from bag B. So if $Y \cap B \neq \emptyset$ then the vertex v is dominated by Y.

Let us assume that there is a node t in T_X with a bag set B that does not contain any vertex from Y. Consider a graph $G_X \setminus B$. If in this graph some vertices of the set X become disconnected we get a contradiction since Y was connecting all vertices from X and $Y \cap B = \emptyset$. So X is connected by Y in $G_X \setminus B$. However, in this case T_X is not an inclusion-minimal subtree with the required property as some connected component of $T_X \setminus \{t\}$ will contain all bags with vertices from the set Y (this follows from Lemma 9). It is not possible that vertices $x_1, x_2 \in X$ belong to bags from different components of $T_X \setminus t$ since the path that connects x_1, x_2 in $G[Y]$ must go through some vertex from bag B and this contradict the fact that $B \cap Y = \emptyset$. □

Lemma 14. *Let G be a chordal graph with n vertices colored in $C = \gamma n$ colors. There is an algorithm that enumerates all* Minimal Tropical Connected Sets *within* $\max_{\alpha:\gamma\le\alpha\le 1} \min\{(\frac{1}{\gamma})^{\gamma n}, 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n}\} \cdot 1.4736^{(1-\alpha+\gamma)n}$ *running time.*

Proof. First of all the algorithm in Lemma 10 constructs a tree-decomposition of graph G in which each bag is a clique. As in the case of general graph we list all potential candidates for the role of the rainbow set R_X. Recall that as before for each tropical set X we associated a rainbow set R_X. The R_X was constructed in the following way: for each $i \in \{1, 2, 3, \dots, C\}$ we put v_i^j in R_X if $v_i^j \in X$ and for each $p < j$ we have that $v_i^p \notin X$. Note that if R_X is a chosen rainbow set in the minimal tropical connected set then vertices v_i^p such that $p < j$ can be deleted from G. Denote by G' the obtained graph after such deletion.

After this, for the fixed rainbow set R_X, the algorithm constructs an R_X-restriction G'_{R_X}. On the next step the algorithm enumerates all minimal connected dominating sets of the graph G'_{R_X}. Let D be a minimal connected dominating set of G'_{R_X}. If $D \cup R_X$ is a minimal tropical connected set we output $D \cup R_X$ (we can test it by Lemma 5).

Let us prove that we output all Minimal Tropical Connected Sets. If Y is Minimal Tropical Connected Set then it contains the associated rainbow subset X' (here it might be the case that $Y = X'$, but it does not contradict anything). Recall that at some point we generate X' as a rainbow set in our algorithm. Since Y is minimal then $S' = Y \setminus X'$ is a minimal set that connects vertices X' (as otherwise there will be a tropical connected set that is a subset of Y). Note that any minimal set that connects X' lies inside $G'_{X'}$ by Lemma 12. For the graph $G'_{X'}$ and sets S', X' conditions of the Lemma 11 are true (take $G'_{X'}$ as G, S' as S and X' as X). Indeed, S' is a minimal set connecting X', $S' \cup X'$ is connected and that is why by Lemma 13 is a dominating set in $G'_{X'}$. Hence, at some point our algorithm considers minimal connected dominating set M' of $G'_{X'}$ such that $S' \subseteq M' \subseteq S' \cup X' = Y$ and outputs $M' \cup X'$ which is exactly Y.

It is left to prove the upper bound on the running time. Construction of the required tree-decomposition of the chordal graph takes polynomial time as well as vertex deletion and construction of R_X-restriction for fixed R_X and G'. So most of the time is consumed by the enumeration of all the rainbow sets and the enumeration of all the connected dominating sets in graph G'_{R_X} for fixed rainbow set R_X. So, as in the case with general graphs, the overall running time up to polynomial factor is:

$$\sum_{\substack{1\le j_1\le n_1\\ \cdots\\ 1\le j_C\le n_C}} 1.4736^{n-(j_1+j_2+\cdots+j_C)+C} = \sum_{\ell:C\le\ell\le n} P_{n,\ell,C}\cdot 1.4736^{n-\ell+C} \le$$

$$n\cdot \max_{\ell:C\le\ell\le n} P_{n,\ell,C}\cdot 1.4736^{n-\ell+C}.$$

By Lemma 4 we know that $P_{n,\ell,C} \le \min\{(\frac{n}{C})^C, \binom{\ell-1}{C-1}\} \le \min\{(\frac{n}{C})^C, \binom{\ell}{C}\}$. Making the substitution $C=\gamma n$ and $\ell=\alpha n$ we have that the running time is at most:

$$\max_{\alpha:\gamma\le\alpha\le 1}\left\{\min\{(\frac{1}{\gamma})^{\gamma n}, 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n}\}\cdot 1.4736^{(1-\alpha+\gamma)n}\right\}.$$

□

Now we have all ingredients to prove the main result of this section.

Theorem 4. *All* MINIMAL TROPICAL CONNECTED SETS *in a chordal graph on n vertices can be enumerated within $\mathcal{O}(1.937^n)$ running time. Hence, the maximum number of* MINIMAL TROPICAL CONNECTED SETS *is at most $\mathcal{O}(1.937^n)$.*

Proof. Our algorithm for chordal graphs as well as the algorithm for the general case combines two algorithms and chooses between them depending on the number of colors in the input graph. However, instead of the algorithm from Lemma 8 we use the algorithm from Lemma 14. We recall that the running time of the algorithm from Lemma 6 is decreasing so it is more suitable for the case when the number of colors $C=\gamma n$ is large. In contrary, the function $\max_{\alpha:\gamma\le\alpha\le 1}\min\{(\frac{1}{\gamma})^{\gamma n}, 2^{H(\frac{\gamma}{\alpha})\cdot\alpha n}\}\cdot 1.4736^{(1-\alpha+\gamma)n}$ is increasing for $\gamma\in[0,\frac{1}{3}]$, see Fig. 2, so the algorithm from Lemma 14 is preferable for small γ. So if $\gamma\le 0.3019$ then we run the algorithm from Lemma 14 and the running time will be bounded by $\mathcal{O}(1.937^n)$. If $\gamma > 0.3019$ then we run the algorithm from Lemma 6 and again the running time will be bounded by $\mathcal{O}(1.937^n)$. □

5 Interval Graphs

The main result of this section improves the previous known upper bound on the maximum number of minimal tropical connected sets in an interval graph on n vertices. Kratsch et al. [27] showed that the number is at most $\mathcal{O}(1.8613^n)$. Our upper bound is $\mathcal{O}(1.7142^n)$. Specifically, we prove the following:

Theorem 5. *There is an algorithm with running time $\mathcal{O}(1.7142^n)$ that enumerates all minimal tropical connected sets in a given interval graph on n vertices. Hence, the number of minimal tropical connected sets in any interval graph is at most $\mathcal{O}(1.7142^n)$.*

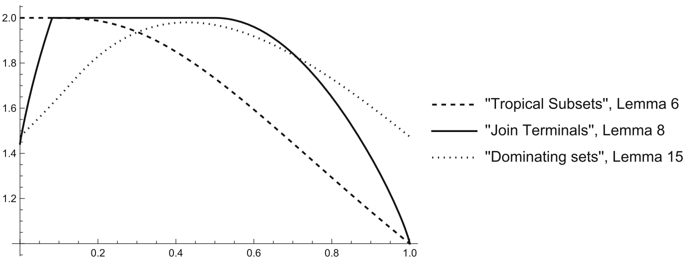

Fig. 2. The dependence of exponent on γ for the algorithms from Lemmas 6, 8, 14

In order to prove the theorem we use the following

Lemma 15. *Given an interval graph G of order n and a subset of vertices $U \subseteq V(G)$ such that $|U| = n'$ we can list all minimal sets Z that connects U in time $\mathcal{O}^*(3^{\frac{n-n'}{3}})$ (i.e. inclusion-minimal sets $Z \subseteq V(G) \setminus U$ such that $G[Z \cup U]$ is a connected graph).*

Proof. As our graph is interval we can in polynomial time construct a interval model such that each vertex has a corresponding interval on a line and:

- two vertices share an edge if and only if corresponding intervals intersect;
- no two intervals share endpoints.

Let us construct such model and fix it. Note that if $G[U]$ is a connected subgraph then the only set Z that satisfies the conditions is $\emptyset$. Denote by $U_1, U_2, \ldots, U_q$ connected components of $G[U]$. We enumerate them from left to right i.e. U_1 is the leftmost connected component in the interval model and U_q is the rightmost connected component. Denote by $\ell(U_i), r(U_i)$ the leftmost and the rightmost point of the connected component U_i on the fixed line model of G.

We must add a few vertices/intervals that join connected components $U_1, \ldots, U_q$. Hence, we must add some vertex whose corresponding interval starts to the left of $r(U_1)$ and ends to the right of $r(U_1)$ as otherwise the connected component U_1 will stay isolated from the other connected components. For a connected subgraph W denote by $N_r(W)$ vertices whose intervals start before $r(W)$ and end to the right of it. For any minimal U-connecting set Z we have $|Z \cap N_r(U_1)| \geq 1$. On the other hand $|Z \cap N_r(U_1)| < 2$, otherwise there are $v_1, v_2 \in Z \cap N_r(U_1)$. If $r(v_1)$ is to the left of $r(v_2)$ then $(Z \setminus v_1)$ connects all components $U_1, \ldots, U_q$ as Z was doing so. If $r(v_1)$ is to the right of $r(v_2)$ then $(Z \setminus v_2)$ connects all components $U_1, \ldots, U_q$. In any case, this contradicts to the fact that Z is inclusion-minimal. So it must be the case that $|Z \cap N_r(U_1)| = 1$.

Based on the above proved facts we suggest the following algorithm:

1. Branch on $|N_r(U_1)|$ possibilities to select a vertex from $N_r(U_1)$ that belongs to Z and discard from the graph the rest of vertices from $N_r(U_1)$. Assume that we pick vertex v' at this step.

2. Run the whole algorithm recursively on the new graph $(G \setminus N_r(U_1)) \cup \{v'\}$ and with a new subset (that needs to be connected) $U \cup \{v'\}$.

The correctness of the presented algorithm follows from the above observations. Since in the recurrence call each time we create i branchings and decrease n by i, we have that the running time of the algorithm is at most $\mathcal{O}^*(3^{\frac{n-n'}{3}})$ (as the maximum of $i^{\frac{1}{i}}$ is achieved when $i = 3$ in the set of natural numbers). □

Now we have all needed tools to present proof of Theorem 5:

Proof. (Proof of Theorem 5). As before we assume that our graph contains vertices of $C = \gamma n$ different colors and the number of vertices colored in the i-th color is exactly n_i, i.e. $n_1 + n_2 + \cdots + n_C = n$. Let $V_i = \{v_i^1, \ldots, v_i^{n_i}\}$ be a set of all vertices of the i-th color.

We know that any tropical set must contain a rainbow set. With each Minimal Tropical Connected Set X we associate a rainbow set R_X constructed as before (for each $i \in \{1, 2, 3, \ldots, C\}$ we put v_i^j in R_X if $v_i^j \in X$ and for each $p < j$ we have that $v_i^j \notin X$). We note that $X \setminus R_X$ is an inclusion-minimal set that connects vertices from R_X, otherwise X is not minimal tropical connected set.

Now we are ready to describe the algorithm. In the first step we list all potential candidates for the role of R_X. So, basically, we consider many branchings and each branch defines a corresponding R_X. So in branch in which $R_X = \{v_1^{j_1}, v_2^{j_2}, \ldots, v_C^{j_C}\}$ we assume that R_X is part of a minimal tropical connected set, while vertices $v_i^{p_i}$ with $p_i < j_i$ are not, hence in this branch these vertices can be simply deleted from the graph. At this point at each branch we already decided about $\ell = j_1 + j_2 + \cdots + j_C$ vertices whether they belong to a minimal tropical connected set or not. There are $n - j_1 - j_2 - \cdots - j_C$ vertices that are left, let us call the set of these vertices W. Now it is enough to list all inclusion-minimal sets $Y' \subseteq W$ such that $R_X \cup Y'$ is connected and discard those sets that are not minimal tropical connected sets. Check whether a set is a minimal tropical connected set can be done in a polynomial time by Lemma 5 and by Lemma 15 we can list all Y' that connect R_X in time $\mathcal{O}^*(3^{\frac{|W|}{3}})$. So the running time of the algorithm up to a polynomial factor is bounded by:

$$\sum_{\substack{1 \le j_1 \le n_1 \\ \cdots \\ 1 \le j_C \le n_C}} 3^{\frac{n-(j_1+j_2+\cdots+j_C)}{3}} \le \sum_{C \le \ell \le n} P_{n,\ell,C} \cdot 3^{\frac{n-\ell}{3}}.$$

Taking into account inequalities from Lemma 4 we have that the running time of our algorithm is at most $poly(n) \cdot \max_\ell[\min\{2^\ell, 3^{\frac{n}{3}}\} \cdot 3^{\frac{n-\ell}{3}}]$. The maximum of previous expression is achieved when $2^\ell = 3^{\frac{n}{3}}$ (since $2^\ell 3^{\frac{n-\ell}{3}}$ is an increasing function of ℓ when n is fixed and $3^{\frac{n}{3}} 3^{\frac{n-\ell}{3}}$ is decreasing). So in the worst case we have $\ell = \frac{n}{3} \cdot \log_2 3$ and the running time of our algorithm is bounded by $\mathcal{O}^*(3^{1/3 \cdot n \cdot (2 - \frac{1}{3} \cdot \frac{\log 3}{\log 2})}) = \mathcal{O}(1.7142^n)$. □

Acknowledgements. We would like to thank Lucas Meijer and anonymous reviewers for comments that helped to improve the paper.

References

1. Agrawal, A., Fomin, F.V., Lokshtanov, D., Saurabh, S., Tale, P.: Path contraction faster than 2^n. SIAM J. Discret. Math. **34**(2), 1302–1325 (2020)
2. 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. Biol. Bioinf. **8**(5), 1296–1308 (2011)
3. Bliznets, I., Fomin, F.V., Pilipczuk, M., Villanger, Y.: Largest chordal and interval subgraphs faster than 2^n. Algorithmica **76**(2), 569–594 (2016)
4. Bliznets, I., Sagunov, D.: Solving target set selection with bounded thresholds faster than 2^n. Algorithmica, 1–22 (2022)
5. Böcker, S., Rasche, F., Steijger, T.: Annotating fragmentation patterns. In: Salzberg, S.L., Warnow, T. (eds.) WABI 2009. LNCS, vol. 5724, pp. 13–24. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04241-6_2
6. Brandstädt, A., Le, V.B., Spinrad, J.P.: Graph classes: a survey. SIAM (1999)
7. Chapelle, M., Cochefert, M., Kratsch, D., Letourneur, R., Liedloff, M.: Exact exponential algorithms to find a tropical connected set of minimum size. In: Cygan, M., Heggernes, P. (eds.) IPEC 2014. LNCS, vol. 8894, pp. 147–158. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13524-3_13
8. Chapelle, M., Cochefert, M., Kratsch, D., Letourneur, R., Liedloff, M.: Exact exponential algorithms to find tropical connected sets of minimum size. Theor. Comput. Sci. **676**, 33–41 (2017)
9. Cohen, J., Italiano, G.F., Manoussakis, Y., Thang, N.K., Pham, H.P.: Tropical paths in vertex-colored graphs. J. Comb. Optim. **42**(3), 476–498 (2021)
10. Cohen, J., Manoussakis, Y., Phong, H., Tuza, Z.: Tropical matchings in vertex-colored graphs. Electron. Notes Discrete Math. **62**, 219–224 (2017)
11. Couturier, J.F., Letourneur, R., Liedloff, M.: On the number of minimal dominating sets on some graph classes. Theoret. Comput. Sci. **562**, 634–642 (2015)
12. Cygan, M., et al.: Parameterized Algorithms. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21275-3
13. Cygan, M., Pilipczuk, M., Pilipczuk, M., Wojtaszczyk, J.O.: Solving the 2-disjoint connected subgraphs problem faster than 2^n. Algorithmica **70**(2), 195–207 (2014)
14. Cygan, M., Pilipczuk, M., Wojtaszczyk, J.O.: Irredundant set faster than $O(2^n)$. In: Calamoneri, T., Diaz, J. (eds.) CIAC 2010. LNCS, vol. 6078, pp. 288–298. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13073-1_26
15. Cygan, M., Pilipczuk, M., Wojtaszczyk, J.O.: Capacitated domination faster than $o(2^n)$. Inf. Process. Lett. **111**(23–24), 1099–1103 (2011)
16. d'Auriac, J.A.A., et al.: Tropical dominating sets in vertex-coloured graphs. J. Discrete Algorithms **48**, 27–41 (2018)
17. d'Auriac, J.A.A., Cohen, N., El Mafthoui, H., Harutyunyan, A., Legay, S., Manoussakis, Y.: Connected tropical subgraphs in vertex-colored graphs. Discrete Math. Theor. Comput. Sci. **17**(3), 327–348 (2016)
18. Fomin, F.V., Gaspers, S., Saurabh, S.: Improved exact algorithms for counting 3- and 4-colorings. In: Lin, G. (ed.) COCOON 2007. LNCS, vol. 4598, pp. 65–74. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73545-8_9
19. Fomin, F.V., Giannopoulou, A.C., Pilipczuk, M.: Computing tree-depth faster than 2^n. Algorithmica **73**(1), 202–216 (2015)
20. Fomin, F.V., Grandoni, F., Kratsch, D.: Solving connected dominating set faster than 2^n. Algorithmica **52**(2), 153–166 (2008)

21. Fomin, F.V., Grandoni, F., Pyatkin, A.V., Stepanov, A.A.: Combinatorial bounds via measure and conquer: bounding minimal dominating sets and applications. ACM Trans. Algorithms (TALG) **5**(1), 1–17 (2008)
22. Fomin, F.V., Kratsch, D.: Exact exponential algorithms (2010)
23. Golovach, P.A., Heggernes, P., Kratsch, D., Saei, R.: Enumeration of minimal connected dominating sets for chordal graphs. Discrete Appl. Math. **278**, 3–11 (2020). https://doi.org/10.1016/j.dam.2019.07.015
24. Golovach, P.A., Kratsch, D., Liedloff, M., Sayadi, M.Y.: Enumeration and maximum number of minimal dominating sets for chordal graphs. Theor. Comput. Sci. **783**, 41–52 (2019). https://doi.org/10.1016/j.tcs.2019.03.017
25. Golumbic, M.C.: Algorithmic Graph Theory and Perfect Graphs. Elsevier, Amsterdam (2004)
26. Heggernes, P.: Minimal triangulations of graphs: a survey. Discret. Math. **306**(3), 297–317 (2006)
27. Kratsch, D., Liedloff, M., Sayadi, M.Y.: Enumerating minimal tropical connected sets. In: Steffen, B., Baier, C., van den Brand, M., Eder, J., Hinchey, M., Margaria, T. (eds.) SOFSEM 2017. LNCS, vol. 10139, pp. 217–228. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-51963-0_17
28. Krzywkowski, M.: Trees having many minimal dominating sets. Inf. Process. Lett. **113**(8), 276–279 (2013)
29. Lacroix, V., Fernandes, C.G., Sagot, M.F.: Motif search in graphs: application to metabolic networks. IEEE/ACM Trans. Comput. Biol. Bioinf. **3**(4), 360–368 (2006)
30. Lawer, E.L.: A note on the complexity of the chromatic number problem. Inf. Process. Lett. (1976)
31. Le, H., Highley, T.: Tropical vertex-disjoint cycles of a vertex-colored digraph: barter exchange with multiple items per agent. Discrete Math. Theor. Comput. Sci. **20** (2018)
32. Moon, J.W., Moser, L.: On cliques in graphs. Israel J. Math. **3**, 23–28 (1965)
33. Razgon, I.: Computing minimum directed feedback vertex set in $o*(1.9977^n)$. In: Theoretical Computer Science, pp. 70–81. World Scientific (2007)
34. Telle, J.A., Villanger, Y.: Connecting terminals and 2-disjoint connected subgraphs. In: Brandstädt, A., Jansen, K., Reischuk, R. (eds.) WG 2013. LNCS, vol. 8165, pp. 418–428. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-45043-3_36
35. Vassilevska Williams, V.: Hardness of easy problems: basing hardness on popular conjectures such as the strong exponential time hypothesis (invited talk). In: 10th International Symposium on Parameterized and Exact Computation (IPEC 2015). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2015)