Parameterized Complexity of Graph Burning

Yasuaki Kobayashi[1] · Yota Otachi[2]

Abstract

GRAPH BURNING asks, given a graph $G = (V, E)$ and an integer k, whether there exists $(b_0, \ldots, b_{k-1}) \in V^k$ such that every vertex in G has distance at most i from some b_i. This problem is known to be NP-complete even on connected caterpillars of maximum degree 3. We study the parameterized complexity of this problem and answer all questions by Kare and Reddy [IWOCA 2019] about the parameterized complexity of the problem. We show that the problem is W[2]-complete parameterized by k and that it does not admit a polynomial kernel parameterized by vertex cover number unless NP ⊆ coNP/poly. We also show that the problem is fixed-parameter tractable parameterized by clique-width plus the maximum diameter among all connected components. This implies the fixed-parameter tractability parameterized by modular-width, by treedepth, and by distance to cographs. Using a different technique, we show that parameterization by distance to split graphs is also tractable. We finally show that the problem parameterized by max leaf number is XP.

Keywords Graph burning · Parameterized complexity · Fixed-parameter tractability

1 Introduction

Bonato, Janssen, and Roshanbin [8, 9] introduced GRAPH BURNING as a model of information spreading. This problem asks to burn all the vertices in a graph in the

✉ Yota Otachi
  otachi@nagoya-u.jp

  Yasuaki Kobayashi
  koba@ist.hokudai.ac.jp

[1] Hokkaido University, Sapporo, Japan

[2] Nagoya University, Nagoya, Japan

following way: we first pick a vertex and set fire to the vertex; at the beginning of each round, the fire spreads one step along edges; at the end of each round, we pick a vertex and set fire to it; the process finishes when all vertices are burned. The objective in the problem is to minimize the number of rounds (including the first round for just picking the first vertex) to burn all the vertices. The minimum number of rounds that can burn a graph G in such a process is the *burning number* of G, which is denoted by $b(G)$. Given a graph G and an integer k, GRAPH BURNING asks whether $b(G) \leq k$.

In other words, the burning number of G can be defined as the minimum length k of a sequence $(b_0, \ldots, b_{k-1})$ of vertices of G such that every vertex in G has distance at most i from some b_i. We call such a sequence a *burning sequence*. Note that in this definition, b_i is the vertex we set fire in the $(k - i)$th round. It is also useful to introduce the generalized neighborhood of vertices. For a vertex v of a graph G, let $N_d[v]$ be the set of vertices with distance at most d in G. For example, $N_0[v]$ contains only v, $N_1[v]$ is the closed neighborhood $N[v]$ of v, and $N_2[v] = \bigcup_{u \in N[v]} N[u]$. With this terminology, a sequence $(b_0, \ldots, b_{k-1})$ of vertices of $G = (V, E)$ is a burning sequence of $G = (V, E)$ if and only if $\bigcup_{0 \leq i \leq k-1} N_i[b_i] = V$.

Note that although GRAPH BURNING is a recent concept, Alon [1] studied the same problem in 1992 as a message transmitting problem on hypercubes motivated by a practical application originated from Intel. Alon [1] showed that the n-dimensional hypercube has burning number $\lceil n/2 \rceil + 1$. This line of research was followed by a few groups of authors [28, 30, 38] and primarily focused on highly symmetric networks.

1.1 Previous Work

As a model of information spreading, it is important to know how fast the information can spread under the model in the worst case. This question can be answered by finding the maximum burning number of graphs of n vertices. The literature is rich in this direction. In the very first papers [8, 9], it is shown that $b(G) \leq 2\lceil \sqrt{n} \rceil - 1$ for every connected graph G of order n and conjectured that $b(G) \leq \lceil \sqrt{n} \rceil$ holds. Note that the connectivity requirement is essential here as an edgeless graph of order n needs n rounds. Some improvements of the general upper bound and studies on special cases are done [3, 6, 7, 11, 15, 20, 27, 34–37, 39, 40, 45], but the conjecture of $b(G) \leq \lceil \sqrt{n} \rceil$ for general connected graphs remains unsettled. The current best upper bound is $\lceil (-3 + \sqrt{24n + 33})/4 \rceil \approx \sqrt{1.5n}$ [34].

The computational complexity of GRAPH BURNING has been studied intensively as well. It is shown that GRAPH BURNING is NP-complete on trees of maximum degree 3, spiders, and linear forests [2]. The NP-completeness result is further extended to connected caterpillars of maximum degree 3 [35], which form subclasses of connected interval graphs, connected permutation graphs, and connected unit disk graphs. On the other hand, GRAPH BURNING admits a 3-approximation algorithm for general graphs [10], that is, given a graph G, the algorithm finds a burning sequence of G with length at most $3 \cdot b(G)$ in polynomial time. Recently, the problem has been shown to be APX-hard [41]. Algorithms with approximation factors parameterized by path-length and tree-length are known as well [31].

Kare and Reddy [32] initiated the study on parameterized complexity of GRAPH BURNING. They showed that GRAPH BURNING on connected graphs is fixed-parameter tractable parameterized by distance to cluster graphs and by neighborhood diversity. The parameterized complexity with respect to the natural parameter k, the burning number, remained open. Recently, Janssen [29] has generalized the problem to directed graphs and has shown that the directed version is W[2]-complete parameterized by k even on directed acyclic graphs. It was mentioned in [29] that the original undirected version parameterized by k was still open.

For further information about the previous studies, see the comprehensive survey by Bonato [5].

1.2 Our Results

In the literature, the input graph of GRAPH BURNING is sometimes assumed to be connected (e.g. in [32]). In GRAPH BURNING, the disconnected case would be nontrivially more complex than the connected case. For example, while the burning number of a path of n vertices is known to be $\lceil \sqrt{n} \rceil$ [8, 9], GRAPH BURNING is NP-complete on disjoint unions of paths [2]. In this paper, we do not assume the connectivity of input graphs. All positive results in this paper hold on possibly disconnected graphs, while all negative results hold even on connected graphs.

Our study in this paper is inspired by Kare and Reddy [32] and Janssen [29]. We generalize the results in [32] and solve all open problems on parameterized complexity in [32]. In Sect. 2, we present fixed-parameter algorithms. We show that GRAPH BURNING is fixed-parameter tractable parameterized by clique-width plus the maximum diameter among all connected components. This implies that GRAPH BURNING is fixed-parameter tractable parameterized by modular-width, by treedepth, and by distance to cographs. We also show that GRAPH BURNING is fixed-parameter tractable parameterized by distance to split graphs. The complexity parameterized by distance to cographs and by distance to split graphs were explicitly asked in [32]. The fixed-parameter tractability parameterized by modular-width generalizes the one parameterized by neighborhood diversity in [32]. In Sect. 3, we present some negative results. We show that GRAPH BURNING parameterized by the natural parameter k is W[2]-complete. This settles the main open problem in this line of research [29, 32]. As a byproduct, we also show that GRAPH BURNING parameterized by vertex cover number does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly. This also answers a question in [32]. Finally, in Sect. 4, we show that GRAPH BURNING is in XP parameterized by the max leaf number and by the maximum diameter among all connected components. See Fig. 1 for a summary of the results.

We assume that the reader is familiar with the basic terms and concepts in the parameterized complexity theory. See some textbooks in the field (e.g., [14, 18]) for definitions. We omit the definitions of most of the graph parameters in this paper as we do not explicitly need them. We only need the definition of the *vertex cover number* of a graph: it is the minimum size of a vertex subset (called a *vertex cover*) such that each edge in the graph has at least one endpoint in the set. We refer the readers to [44] for the definitions of other graph parameters and the hierarchy among them.
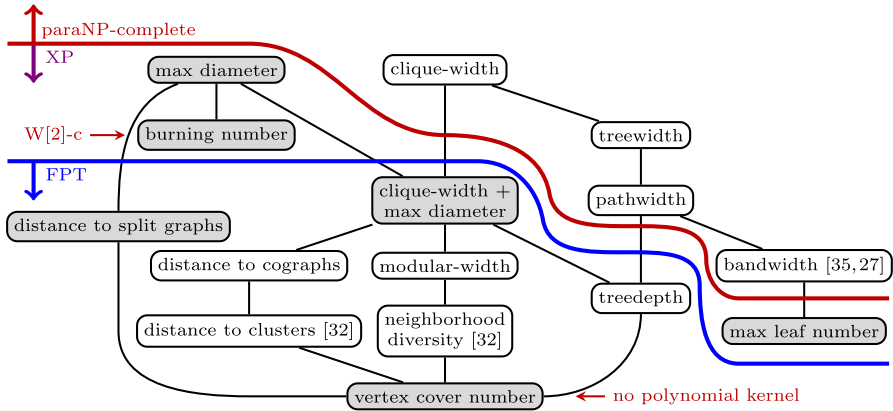
Fig. 1 Graph parameters and the complexity of GRAPH BURNING. Our results in this paper are on the parameters with the dark background. Connections between two parameters imply the existence of a function in the one above (being in this sense more general) that lower-bounds the one below. "The maximum diameter of the components" is shortened as "max diameter"

2 Fixed-Parameter Tractability

We first observe that GRAPH BURNING is expressible as a first order logic (FO) formula of length depending only on k.

The syntax of FO of graphs includes (i) the logical connectives $\vee, \wedge, \neg, \Leftrightarrow, \Rightarrow$, (ii) variables for vertices, (iii) the quantifiers $\forall$ and $\exists$ applicable to these variables, and (iv) the following binary relations: equality of variables, and $\mathsf{adj}(u, v)$ for two vertex variables u and v, which means that u and v are adjacent. If G models an FO formula φ with no free variables, then we write $G \models \varphi$.

The following formula $\mathrm{dist}_{\leq d}(v, w)$ is true if and only if the distance between v and w is at most d:

$$\mathrm{dist}_{\leq d}(v, w) := \exists u_0, \ldots, u_d \; (u_0 = v) \wedge (u_d = w)$$

$$\wedge \left(\bigwedge_{0 \leq i < d} (u_i = u_{i+1}) \vee \mathsf{adj}(u_i, u_{i+1}) \right).$$

Clearly, $\mathrm{dist}_{\leq d}(v, w)$ has length depending only on d. Now we define the formula φ_k such that $G \models \varphi_k$ if and only if (G, k) is a yes instance of GRAPH BURNING as follows:

$$\varphi_k := \exists v_0, \ldots, v_{k-1} \; \forall u \bigvee_{0 \leq i \leq k-1} \mathrm{dist}_{\leq i}(u, v_i).$$

It is known that on nowhere dense graph classes, testing an FO formula ψ is fixed-parameter tractable parameterized by $|\psi|$, where $|\psi|$ is the length of ψ [25]. (See [25] for the definition of nowhere dense graph classes.) Since φ_k is an FO formula of length depending only on k, the following holds.

Observation 2.1 GRAPH BURNING on nowhere dense graph classes parameterized by k is fixed-parameter tractable.

For an n-vertex graph G of clique-width at most cw and for a one-sorted monadic-second order logic formula (an MSO_1 formula, for short) ψ, one can check whether $G \models \psi$ in time $O(f(|\psi|, \mathsf{cw}) \cdot n^3)$, where f is a computable function [13, 43]. Since an FO formula is an MSO_1 formula and the length of φ_k depends only on k, we can observe the following fact.

Observation 2.2 GRAPH BURNING parameterized by clique-width $+ k$ is fixed-parameter tractable.

We extend this observation in a nontrivial way to show the main result of this section. To this end, it is useful to generalize φ_k as follows. For nonempty $I = \{i_1, \dots, i_{|I|}\} \subseteq \{0, \dots, k-1\}$, let φ_I be a formula that means that there are vertices $b_{i_1}, \dots, b_{i_{|I|}}$ such that $\bigcup_{i_j \in I} N_{i_j}[b_{i_j}] = V$, which can be expressed as follows:

$$\varphi_I := \exists v_{i_1}, \dots, v_{i_{|I|}} \ \forall u \bigvee_{1 \leq j \leq |I|} \mathrm{dist}_{\leq i_j}(u, v_{i_j}).$$

Clearly, checking $G \models \varphi_I$ is still fixed-parameter tractable parameterized by clique-width $+ k$.

Theorem 2.3 GRAPH BURNING *is fixed-parameter tractable parameterized by clique-width plus the maximum diameter among all connected components.*

Proof Let (G, k) be an instance of GRAPH BURNING, where G has n vertices. Let $C_1, \dots, C_p$ be the connected components of G and $d_{\max}$ be the maximum diameter of the components. We assume that $k \geq p$ since otherwise (G, k) is a trivial no instance. By Observation 2.2, we can also assume that $d_{\max} < k$.

Observe that if a component C_q contains b_i for some $i \geq d_{\max}$, then $N_i[b_i] = V(C_q)$. Hence the problem is equivalent to finding a sequence $(b_0, \dots, b_{d_{\max}-1})$ that burns as many connected components as possible. That is, we want to find a maximum cardinality subset $\mathcal{C} \subseteq \{C_1, \dots, C_p\}$ and a sequence $(b_0, \dots, b_{d_{\max}-1})$ such that $\bigcup_{0 \leq i \leq d_{\max}-1} N_i[b_i] = \bigcup_{C_q \in \mathcal{C}} V(C_q)$. It holds that $p - |\mathcal{C}| \leq k - d_{\max}$ if and only if (G, k) is a yes instance.

We reduce this problem to DISJOINT SETS. Given a universe U, a subset family $\mathcal{S} \subseteq 2^U$, and an integer t, DISJOINT SETS asks whether there are t pairwise-disjoint subsets in $\mathcal{S}$. Let $U = \{0, 1, \dots, d_{\max} - 1\} \cup \{c_1, \dots, c_p\}$ and $\mathcal{S} = \{I \cup \{c_q\} \mid I \subseteq \{0, \dots, d_{\max} - 1\}, C_q \models \varphi_I, 1 \leq q \leq p\}$. Clearly, picking $I \cup \{c_q\}$ into the solution for the DISJOINT SET instance corresponds to burning C_q with $\{b_i \mid i \in I\}$, and vice versa. We set $t = p - k + d_{\max}$. Note that $t \leq d_{\max}$ as $k \geq p$. Using the color-coding technique, it can be shown that DISJOINT SETS is solvable in time $2^{O(t \cdot \max_{S \in \mathcal{S}} |S|)}(|\mathcal{S}| + |U|)^{O(1)}$ [18, DISJOINT r- SUBSETS] (see also [16, BOUNDED RANK DISJOINT SETS]). In our instance, $t \cdot \max_{S \in \mathcal{S}} |S| \leq d_{\max}(d_{\max} + 1)$ holds.

For each $q \in \{1, \dots, p\}$ and for each $I \subseteq \{0, \dots, d_{\max} - 1\}$, we can test whether $C_q \models \varphi_I$ in time $O(f(d_{\max} + \mathsf{cw}) \cdot n^3)$ for some computable function f, where cw is

the clique-width of G, because $|\varphi_I|$ depends only on $d_{\max}$. Thus, $\mathcal{S}$ can be constructed in time $O(f(d_{\max} + \mathsf{cw}) \cdot n^3 \cdot p \cdot 2^{d_{\max}})$. Since $|\mathcal{S}| \leq 2^{d_{\max}} \cdot p$, the last step of solving DISJOINT SETS can be done in time $2^{O(d_{\max}^2)}(2^{d_{\max}} \cdot p + (d_{\max} + p))^{O(1)}$. Since $p \leq n$, the total running time is $g(d_{\max} + \mathsf{cw}) \cdot n^{O(1)}$ for some computable function g. This completes the proof. □

The definition of modular-width [23] implies that every connected component of a graph of modular-width at most w has both diameter and clique-width at most w. It is known that every connected component of a graph of treedepth at most d has diameter at most 2^d [42] and its clique-width is bounded by a function of treedepth (or even smaller treewidth) [12]. Therefore, Theorem 2.3 implies the fixed-parameter tractability with respect to these parameters.

Corollary 2.4 GRAPH BURNING *is fixed-parameter tractable parameterized by modular-width.*

Corollary 2.5 GRAPH BURNING *is fixed-parameter tractable parameterized by treedepth.*

For a graph class $\mathcal{C}$ and a graph G, the *distance* from G to $\mathcal{C}$ is defined as the minimum integer k such that by removing at most k vertices from G, one can obtain a member of $\mathcal{C}$.

Let $\mathcal{C}$ be a graph class with constants c and d such that each graph in $\mathcal{C}$ has clique-width at most c and each connected component of each member of $\mathcal{C}$ has diameter at most d. Observe that a graph of distance at most k to $\mathcal{C}$ has clique-width at most $c \cdot 2^k$ since after a removal of a single vertex, the clique-width remains at least half of the original clique-width [26]. Observe also that each connected component of a graph of distance at most k to $\mathcal{C}$ has diameter less than $(k+1)(d+2)$. Thus, by Theorem 2.3, GRAPH BURNING is fixed-parameter tractable parameterized by distance to $\mathcal{C}$. This observation can be applied immediately to cographs that are known to be the P_4-free graphs and the graphs of clique-width at most 2.

Corollary 2.6 GRAPH BURNING *is fixed-parameter tractable parameterized by distance to cographs.*

Now we consider the distance to split graphs. A graph is a *split graph* if its vertex set can be partitioned into a clique and an independent set. For a graph $G = (V, E)$, which is not necessarily a split graph, a subset $S \subseteq V$ is a *split-deletion set* if $G - S$ is a split graph. Then the distance to split graphs from G is equal to the minimum size of a split-deletion set. It is known that the split graphs are exactly the $(2K_2, C_4, C_5)$-free graphs [21]. This characterization implies that when designing an algorithm parameterized by distance d to split graphs, we can assume that a split-deletion set of minimum size is given since a standard bounded-search tree algorithm finds such a set in time $5^d \cdot n^{O(1)}$, where 5 is the maximum order of the forbidden induced subgraphs.

Theorem 2.7 GRAPH BURNING *is fixed-parameter tractable parameterized by distance to split graphs.*

Fig. 2 The partition of the vertex set in the proof of Theorem 2.7
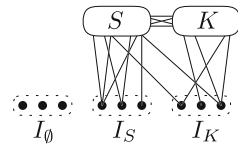
Proof Let (G, k) be an instance of GRAPH BURNING and S be a minimum split-deletion set of $G = (V, E)$. We denote $|S|$ by s. Let (K, I) be a partition of $V - S$, where K is a clique and I is an independent set. It is well known that such a partition can be found in linear time by greedily adding a vertex of minimum degree in $G - S$ into I. We further partition I into three subsets I_K, I_S, $I_\emptyset$ in such a way that $I_\emptyset$ is the set of degree-0 vertices in G, I_S is the set of vertices in $I \setminus I_\emptyset$ that have neighbors only in S, and $I_K = I \setminus (I_\emptyset \cup I_S)$. Note that each vertex in I_S has at least one neighbor in S and each vertex in I_K has at least one neighbor in K (and possibly some neighbors in S). See Fig. 2.

First observe that if $k \geq s + |I_\emptyset| + 3$, then (G, k) is a yes instance: take two vertices arbitrarily as b_0 and b_1, one vertex in K as b_2, and all the vertices in $S \cup I_\emptyset$ as the rest of a burning sequence. Hence, in the following, we assume that $k < s + |I_\emptyset| + 3$.

We now reduce the number of vertices in I_S. For a nonempty subset $S' \subseteq S$, let $J_{S'} \subseteq I_S$ be the set of vertices whose neighborhood is exactly S'. Since the pairwise distance between vertices in $J_{S'}$ is 2, a burning sequence does not need four or more vertices in $J_{S'}$. Thus, we can remove all but three vertices in $J_{S'}$ and obtain an equivalent instance. We apply this reduction to all subsets $S' \subseteq S$ and denote the reduced subset of I_S by I_S^*. Note that $|I_S^*| < 3 \cdot 2^{|S|}$.

We then remove all vertices in $I_\emptyset$ and obtain an equivalence instance of a slightly generalized problem. Observe that if (G, k) is a yes instance, then there is a burning sequence $(b_0, \ldots, b_{k-1})$ of G such that the set of first $|I_\emptyset|$ vertices $\{b_0, \ldots, b_{|I_\emptyset|-1}\}$ is $I_\emptyset$: we need to take every isolated vertex into a burning sequence, but even b_0 is good enough to burn an isolated vertex. Hence, the problem can be reduced to the one for finding a sequence $(b_{|I_\emptyset|}, b_{|I_\emptyset|+1}, \ldots, b_{k-1})$ of vertices in $V \setminus I_\emptyset$ such that $\bigcup_{|I_\emptyset| \leq i \leq k-1} N_i[b_i] = V \setminus I_\emptyset$. We denote by $(G', k, |I_\emptyset|)$ the obtained instance of the new problem, where $G' = G[K \cup S \cup I_S^*]$.

To solve the reduced problem, we first guess which vertices in $S \cup I_S^*$ appear in $(b_{|I_\emptyset|}, b_{|I_\emptyset|+1}, \ldots, b_{k-1})$ and where they are placed in the sequence. The number of candidates of such a guess depends only on s as $|S \cup I_S^*|^{k-|I_\emptyset|} < (s + 3 \cdot 2^s)^{s+3}$. The vacant slots of $(b_{|I_\emptyset|}, b_{|I_\emptyset|+1}, \ldots, b_{k-1})$ after the guess tell us which b_i belongs to $K \cup I_K$. If there are at most three vertices in $K \cup I_K$ that appear in $(b_{|I_\emptyset|}, b_{|I_\emptyset|+1}, \ldots, b_{k-1})$, then we try all $O(n^3)$ combinations to complete the sequence. Otherwise, we guess from $O(n)$ candidates the vertex in $K \cup I_K$ that appears in $(b_{|I_\emptyset|}, b_{|I_\emptyset|+1}, \ldots, b_{k-1})$ and has the largest index. Since the index of the guessed vertex in $(b_{|I_\emptyset|}, b_{|I_\emptyset|+1}, \ldots, b_{k-1})$ is at least 3 and $K \cup I_K$ induces a connected split graph, which has diameter at most 3, the guessed vertex in $K \cup I_K$ burns all vertices in $K \cup I_K$.

Finally we fill the positions in $(b_{|I_\emptyset|}, b_{|I_\emptyset|+1}, \ldots, b_{k-1})$ that still remain vacant. Let $X \subseteq \{|I_\emptyset|, \ldots, k-1\}$ be the set of indices i for which no vertex is guessed as b_i so far, and let $\overline{X} = \{|I_\emptyset|, \ldots, k-1\} \setminus X$. Let $U = V(G') \setminus (\bigcup_{i \in \overline{X}} N_i[b_i])$. Note that

$U \subseteq S \cup I_S^*$. Our task is to find $\{b_i \mid i \in X\} \subseteq K \cup I_K$ such that $U \subseteq \bigcup_{i \in X} N_i[b_i]$. This task can be seen as an instance $(U', \mathcal{S}, p)$ of SET COVER, where $U' = U \cup X$, $\mathcal{S} = \{(N_i[v] \cap U) \cup \{i\} \mid v \in K \cup I_K, i \in X\}$, and $p = |X| < k - |I_\emptyset| < s + 3$. Since SET COVER parameterized by $|U'|$ is fixed-parameter tractable [22,Lemma 2] and $|U'| \leq s + 3 \cdot 2^s + p$, the theorem follows. □

3 Fixed-Parameter Intractability

This section is devoted to the proofs of the following theorems.

Theorem 3.1 GRAPH BURNING *is* W[2]-*complete parameterized by* k.

Theorem 3.2 GRAPH BURNING *does not admit a polynomial kernel parameterized by vertex cover number unless* NP $\subseteq$ coNP/poly.

We present a reduction from SET COVER to GRAPH BURNING that proves both Theorems 3.1 and 3.2. Given a set $U = \{u_1, \ldots, u_n\}$, a family of nonempty subsets $\mathcal{S} = \{S_1, \ldots, S_m\} \subseteq 2^U \setminus \{\emptyset\}$, and a positive integer s, SET COVER asks whether there exists a subfamily $\mathcal{S}' \subseteq \mathcal{S}$ such that $|\mathcal{S}'| \leq s$ and $\bigcup_{S \in \mathcal{S}'} S = U$.

Let $(U = \{u_1, \ldots, u_n\}, \mathcal{S} = \{S_1, \ldots, S_m\}, s)$ be an instance of SET COVER. We construct an equivalent instance $(G, k = s + 2)$ of GRAPH BURNING. (See Fig. 3.) We first construct $s = k - 2$ isomorphic graphs $G_2, \ldots, G_{k-1}$ as follows. For each $i \in \{2, 3, \ldots, k-1\}$, the vertex set of G_i is $U_i \cup V_i$, where $U_i = \{u_1^{(i)}, \ldots, u_n^{(i)}\}$ is a clique and $V_i = \{v_1^{(i)}, \ldots, v_m^{(i)}\}$ is an independent set. In G_i, $u_p^{(i)}$ and $v_q^{(i)}$ are adjacent if and only if $u_p \in S_q$. From each G_i, we construct H_i by adding $i + 2$ copies of a path of i vertices and all possible edges between each vertex in V_i and one of the degree-1 vertices in each path. We then take the disjoint union of $H_2, H_3, \ldots, H_{k-1}$ and add U as a clique. For each $i \in \{2, 3, \ldots, k-1\}$ and $j \in \{1, 2, \ldots, m\}$, we connect $u_j^{(i)}$ and u_j with a path of length $i - 1$ with $i - 2$ new inner vertices. Finally, we attach a vertex w to a vertex in U, and a path (x, y, z) to the same vertex. We set $V_0 = \{w\}$ and $V_1 = \{x, y, z\}$. We denote the constructed graph by G.

Lemma 3.3 $(U, \mathcal{S}, s)$ *is a yes instance of* SET COVER *if and only if* (G, k) *is a yes instance of* GRAPH BURNING.

Proof ($\Longrightarrow$) Assume that $(U, \mathcal{S}, s)$ is a yes instance of SET COVER and $\mathcal{S}' \subseteq \mathcal{S}$ is a certificate; that is, $|\mathcal{S}'| \leq s$ and $\bigcup_{S \in \mathcal{S}'} S = U$. We assume without loss of generality that $|\mathcal{S}'| = s$ and $\mathcal{S}' = \{S_2, S_3, \ldots, S_{s+1=k-1}\}$. We set $b_0 = w$, $b_1 = y$, and $b_i = v_i^{(i)}$ for $2 \leq i \leq k-1$. We show that $(b_0, \ldots, b_{k-1})$ is a burning sequence of G.

Clearly, $N_0[b_0] = V_0$ and $N_1[b_1] = V_1$. For $2 \leq i \leq k-1$, observe that $N_i[b_i] = N_i[v_i^{(i)}]$ includes all the vertices of H_i: the farthest vertices in i-vertex paths have distance exactly i from $v_i^{(i)}$; $\mathrm{dist}(v_i^{(i)}, v_j^{(i)}) = 2$ for each $j \neq i$ as $v_i^{(i)}$ and $v_j^{(i)}$ share a neighbor (an endpoint of a path of i vertices); $\mathrm{dist}(v_i^{(i)}, u_j^{(i)}) \leq 2$ for every j since $v_i^{(i)}$ has at least one neighbor in the clique U_i as $\emptyset \notin \mathcal{S}$. Moreover, $N_i[v_i^{(i)}]$ includes all inner vertices of the paths from U_i to U as $\mathrm{dist}(v_i^{(i)}, u_j^{(i)}) \leq 2$ for every j. Finally,
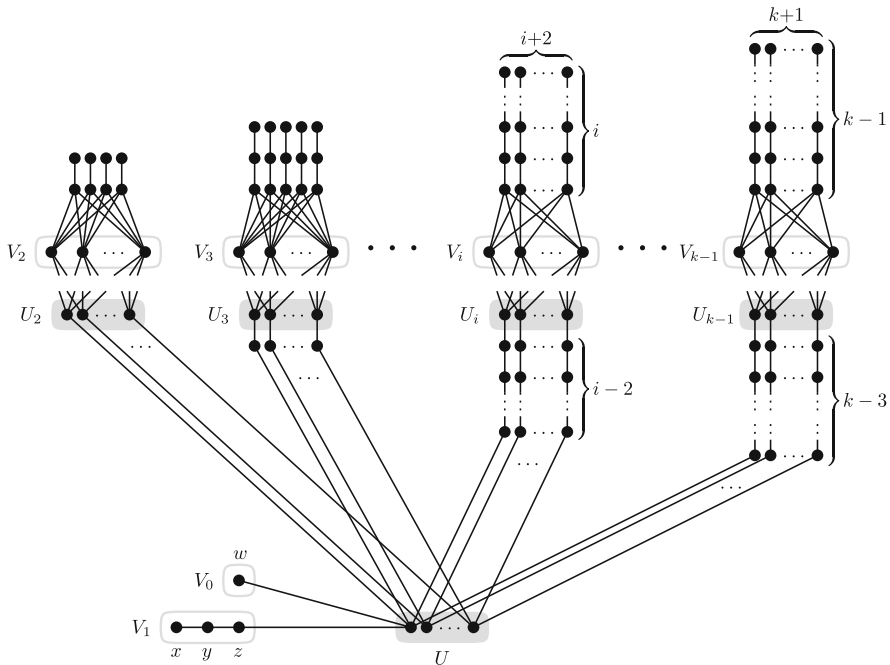
Fig. 3 The reduction from SET COVER to GRAPH BURNING. The edges in the cliques $U_2, \ldots, U_{k-1}$, and U are omitted

$u_j \in N_i[v_i^{(i)}]$ if and only if $u_j \in S_i$: if $u_j \in S_i$, then $v_i^{(i)}$ and $u_j^{(i)}$ are adjacent, and thus $\mathrm{dist}(v_i^{(i)}, u_j) \leq 1 + \mathrm{dist}(u_j^{(i)}, u_j) = i$; otherwise, $v_i^{(i)}$ and $u_j^{(i)}$ are not adjacent and thus $\mathrm{dist}(v_i^{(i)}, u_j) \geq 1 + \mathrm{dist}(u_{h \neq j}^{(i)}, u_j) > i$. This implies that $U \subseteq \bigcup_{2 \leq i \leq k-1} N_i[v_i^{(i)}]$ since $\bigcup_{S \in \mathcal{S}'} S = U$.

($\Longleftarrow$) Assume that (G, k) is a yes instance of GRAPH BURNING and $(b_0, \ldots, b_{k-1})$ is a burning sequence of G.

We first show that $b_i \in V_i$ for all $0 \leq i \leq k-1$. Let $i \in \{2, \ldots, k-1\}$. Assume that we already know that $b_j \in V_j$ for $i + 1 \leq j \leq k - 1$. Since there are $i + 2$ paths attached to V_i, at least one of them, say P, has no vertex in the remaining vertices $b_0, \ldots, b_i$. The degree-1 vertex in P has distance exactly i from every vertex in V_i and distance at least $i + 1$ from every vertex not in $V(P) \cup V_i$. Hence, $b_i \in V_i$. Now we know that $b_i \in V_i$ for $2 \leq i \leq k - 1$. Let $u_j \in U$ be the vertex where V_0 and V_1 are attached to. For $2 \leq i \leq k - 1$, we have $\mathrm{dist}(b_i, u_j) \geq i$, and thus $N_i[b_i]$ contains no vertex in $V_0 \cup V_1$. Since b_0 covers only b_0 itself, $b_1 = y$ and $b_0 = w$ hold.

For $2 \leq i \leq k - 1$, let $b_i = v_{h_i}^{(i)}$. Since $N_0[b_0] = \{w\}$ and $N_1[b_1] = \{x, y, z\}$, we have $U \subseteq \bigcup_{2 \leq i \leq k-1} N_i[v_{h_i}^{(i)}]$. As we saw in the only-if case, $u_j \in N_i[v_{h_i}^{(i)}]$ if and only if $u_j \in S_{h_i}$ for $1 \leq j \leq n$. This implies that $N_i[v_{h_i}^{(i)}] \cap U = S_{h_i}$, and thus $U = \bigcup_{2 \leq i \leq k-1} S_{h_i}$. Therefore, the subfamily $\{S_{h_2}, S_{h_3}, \ldots, S_{h_{k-1}}\} \subseteq \mathcal{S}$ of at most $k - 2 = s$ subsets shows that $(U, \mathcal{S}, s)$ is a yes-instance of SET COVER. $\qquad\square$

Proof *(Theorem 3.1)* By Lemma 3.3, the construction of (G, k) from $(U, \mathcal{S}, s)$ described above is a parameterized reduction from SET COVER parameterized by s to GRAPH BURNING parameterized by $k = s + 2$. Since SET COVER is W[2]-complete parameterized by s [17], the W[2]-hardness follows.

The membership to W[2] can be shown by the following reduction to SET COVER. Let $(G = (V, E), k)$ be an instance of GRAPH BURNING. We set $s = k$, $U = V \cup \{0, 1, \ldots, k - 1\}$, and $\mathcal{S} = \{N_i[v] \cup \{i\} \mid v \in V, 0 \leq i \leq k - 1\}$. This is just an undirected version of the proof by Janssen [29], who showed the membership to W[2] for GRAPH BURNING on directed graphs, and the correctness can be shown in the same way. □

Proof *(Theorem 3.2)* The graph G constructed above has an independent set $\bigcup_{i=2}^{k-1} V_i$. The vertices not belonging to this independent set form a vertex cover of size $4 + (k - 1)|U| + \sum_{2 \leq i \leq k-1}(i(i+2)+(i-2)|U|)$, which is a polynomial in $k = s+2$ and $|U|$. By Lemma 3.3, the construction of (G, k) from $(U, \mathcal{S}, s)$ described above is a polynomial parameter transformation [4] from SET COVER parameterized by $|U| + s$ to GRAPH BURNING parameterized by vertex cover number. Since SET COVER parameterized by $|U| + s$ does not admit polynomial kernels unless NP $\subseteq$ coNP/poly [16], the theorem holds. □

The reduction above also shows the W[2]-hardness parameterized by diameter since the diameter of a connected graph is smaller than the square of its burning number [8, 9]. We further observe that the graph G in the reduction is $P_{(4k-3)}$-free. That is, G does not contain a path of $4k - 3$ vertices as an induced subgraph. Let P be an induced path in G. For $i \in \{2, \ldots, k - 1\}$, let H_i' be the graph consists of H_i and the paths from U_i to U. Since U is a clique, there are at most two indices i such that P intersects $H_i' - U$. We can see that $|V(P) \cap V(H_i')| \leq 2i + 1$ for each i, and thus $|V(P)| \leq 2(k - 1) + 1 + 2(k - 2) + 1 = 4k - 4$. This implies the following W[2]-hardness.

Corollary 3.4 GRAPH BURNING *on P_q-free graphs is* W[2]-*hard parameterized by q.*

4 XP Algorithms

To complete the picture of parameterized complexity of GRAPH BURNING (Fig. 1), we present a couple of XP algorithms in this section. Recall that a parameterized problem is *slice-wise polynomial* (or XP, for short) if it admits an algorithm of running time $O(n^{f(k)})$ for some computable function f, where n is the input size and k is the parameter.

We saw in the previous section that GRAPH BURNING is W[2]-hard parameterized by the maximum diameter of the components. The following simple observation complements this fact by placing it into XP.

Observation 4.1 GRAPH BURNING is XP parameterized by the maximum diameter among all connected components.

Proof Let (G, k) be an instance of GRAPH BURNING, n be the number of vertices in G, and $d_{\max}$ be the maximum diameter of the components of G. As observed in the proof of Theorem 2.3, the problem is equivalent to finding a sequence $(b_0, \ldots, b_{d_{\max}-1})$ that burns as many connected components as possible. We try all possible sequences of $d_{\max}$ vertices in G and find the one that burns the maximum number of components. Since there are at most $n^{d_{\max}}$ such sequences, this algorithm runs in time $n^{O(d_{\max})}$. □

We now turn to the main result of this section, which is on max leaf number. The *max leaf number* of a connected graph is the maximum integer ℓ such that the graph has a spanning tree with ℓ leaves. For disconnected graphs, the max leaf number is the sum of those of its connected components. It is known that the bandwidth of a graph is upper-bounded by the twice of its max leaf number [44]. Since GRAPH BURNING is NP-complete already on caterpillars of maximum degree 3 [27, 35], which have bandwidth at most 2, it would be important to see how the problem changes its complexity when parameterized by max leaf number.

In the rest of this section, we show that the burning number of G can be computed in time $n^{O(\ell^2)}$, where ℓ is the max leaf number of G, yielding that GRAPH BURNING is XP parameterized by max leaf number.

Theorem 4.2 GRAPH BURNING *is XP parameterized by max leaf number.*

Proof Let (G, k) be an instance of GRAPH BURNING, where $G = (V, E)$ is an n-vertex graph of max leaf number ℓ. It is known that such a graph is a subdivision of a graph H with at most $4\ell - 2$ vertices [19, 33]. Thus, the edge set E of G can be partitioned into edge-disjoint paths $P_1, \ldots, P_p$ with $p \leq \binom{4\ell-2}{2}$ such that for $1 \leq i \leq p$, every internal vertex of P_i has degree exactly 2 in G. We call each end vertex of P_i a *connection point*. Note that such a partition with the minimum number of paths can be computed in polynomial time (while computing a spanning tree with ℓ leaves is NP-hard [24]). Let $\mathcal{P} = \{P_1, \ldots, P_p\}$ be such a partition.

In the following, we assume that the burning number of G is at least $(4\ell-2)(4\ell-3)$ as otherwise we can compute $\mathsf{b}(G)$ in time $n^{O(\ell^2)}$ by checking all possible burning sequences. For a burning sequence $B = (b_0, \ldots, b_{k-1})$ of G, we call b_i the *ith source* of B for each $0 \leq i \leq k-1$.

Let $B = (b_0, \ldots, b_{k-1})$ be a burning sequence of G and let $P = (v_1, \ldots, v_t)$ be a path in G whose internal vertices have degree exactly 2 in G. Assume that B contains at least one vertex $v_{i'}$ of P as the ith source of B (i.e., $b_i = v_{i'}$). We say that b_i is *left-effective for P (with respect to B)* if for every $v_{j'}$ contained in B as the jth source (i.e., $b_j = v_{j'}$), it holds that $i - i' \geq j - j'$. Similarly, b_i is *right-effective for P (with respect to B)* if it satisfies $i + i' \geq j + j'$ for any $v_{j'}$ contained in B as the jth source (i.e., $b_j = v_{j'}$). Intuitively, if the fire starting at a vertex in P burns a vertex $u \notin V(P)$, then at least one of the left- and right-effective vertices for P burns u as well. If P contains at least one vertex b_i in B, it also contains (possibly identical) left- and right-effective vertices with respect to B. Note that for a path P, there can be more than one left-effective vertices (and also right-effective vertices). Let x and y be (possibly identical) left- and right-effective vertices for P, respectively. Observe that for every vertex $z \in (V(B) \cap V(P)) \setminus \{x, y\}$ and $v \in V(G) \setminus V(P)$, if v is burned by

z, then v is also burned by at least one of x and y. This simple observation plays a key role in the following discussion.

The idea of our algorithm is as follows. For each path $P \in \mathcal{P}$, we guess at most two vertices of a burning sequence $B = (b_0, \ldots, b_{k-1})$ for its left- and right-effective vertices. If these guesses are correct for B, the vertices "unburned" by the effective vertices induce a disjoint union of $O(\ell^2)$ paths, and the remaining problem is easy to solve by dynamic programming.

To be precise, we first select a subset S of vertices of G such that for each $P \in \mathcal{P}$, $|V(P) \cap S| \le 2$. Clearly, we have $|S| \le (4\ell - 2)(4\ell - 3)$ and hence the number of possible subsets is $n^{O(\ell^2)}$. For such S, we also consider an assignment to a burning sequence, which is defined by an injection γ from S to $\{0, 1, \ldots, k-1\}$. We say that (S, γ) is *correct* for B if (1) for every $P \in \mathcal{P}$ with $V(P) \cap S \ne \emptyset$, S contains at least one left-effective vertices and at least one right-effective vertices for P, and (2) every vertex in S is left- or right-effective for some $P \in \mathcal{P}$. The key to our XP algorithm is the following claim.

Claim Let $B = (b_0, \ldots, b_{k-1})$ be a burning sequence of G, $S \subseteq V(B)$, and $\gamma : S \to \{0, 1, \ldots, k-1\}$. If (S, γ) is correct for B, then each component in $G[V \setminus (\bigcup_{v \in S} N_{\gamma(v)}[v])]$ is a subpath of P for some $P \in \mathcal{P}$ that has no connection points.

Proof (Claim) We show that each connection point u is burned by some vertex in S under assignment γ, that is, $u \in \bigcup_{v \in S} N_{\gamma(v)}[v]$. Suppose to the contrary that $u \notin \bigcup_{v \in S} N_{\gamma(v)}[v]$ for a connection point u on a path $P \in \mathcal{P}$. As B is a burning sequence of G, there is $b_i \in V(B)$ with $u \in N_i[b_i]$. Let $P' \in \mathcal{P}$ be the path containing b_i. Then, S contains at least one left-effective vertex and at least one right-effective vertex for P'. By the definition of effective vertices, at least one of them also burns u, contradicting to the choice of u. Thus, $\bigcup_{v \in S} N_{\gamma(v)}[v]$ contains all connection points in G and hence each remaining component is a subpath of P. $\qquad \square$

The claim above ensures that if (S, γ) is correct for a burning sequence $B = (b_0, \ldots, b_{k-1})$, then the set of the remaining vertices $R := V \setminus (\bigcup_{v \in S} N_{\gamma(v)}[v])$ induces a disjoint union of paths. Moreover, as $|S \cap V(P)| \le 2$ for each $P \in \mathcal{P}$, there are at most $3p$ paths in $G[R]$. Let $\mathcal{R} = \{P_1', \ldots, P_{p'}'\}$ be the vertex-disjoint paths that form $G[R]$. The remaining task is to burn those paths by appropriately selecting the ith source of B for each $i \in \{0, \ldots, k-1\} \setminus \{\gamma(v) \mid v \in S\}$. Since S contains left- and right-effective vertices for each path in $\mathcal{P}$ (if they exist), the fire starting at a vertex on a path $P \in \mathcal{R}$ does not spread to a vertex on another path $P' \in \mathcal{R}$. Thus, we can assume that the ith source of B is selected from R for each $i \in \{0, \ldots, k-1\} \setminus \{\gamma(v) \mid v \in S\}$.

To solve the remaining task, it suffices to give an algorithm that given a disjoint union of paths $P_1, \ldots, P_t$ and a set of indices $I \subseteq \{0, \ldots, k-1\}$, decides if there is a set $\{b_i \mid i \in I\}$ of $|I|$ vertices such that $\bigcup_{i \in I} N_i[b_i]$ covers all vertices in the paths. If such a covering exists, then we say that the pair $(I, \{P_1, \ldots, P_t\})$ is *feasible*. For each $1 \le i \le |I|$, we denote by q_i the ith smallest element in I and by I_i the set of smallest i elements in I, i.e., $I_i = \{q_1, \ldots, q_i\}$. For each $1 \le i \le t$, let n_i be the number of vertices in P_i. Let $\mathcal{P} = \{P_1, \ldots, P_t\}$. Observe that $(I, \mathcal{P})$ is feasible if and only if so is $(I_{|I|-1}, \mathcal{P}_i)$ for some $i \in \{1, \ldots, t\}$, where $\mathcal{P}_i$ is obtained from $\mathcal{P}$

by replacing the path $P_i \in \mathcal{P}$ with a path of $n_i - 2q_{|I|} - 1$ vertices as $b_{q_{|I|}}$ can cover $2q_{|I|} + 1$ consecutive vertices in a path. (If $n_i - 2q_{|I|} - 1 \le 0$, then we just remove P_i.) This observation leads to the following dynamic programming algorithm. We compute $\mathsf{opt}(i; \ell_1, \ldots, \ell_t)$ that indicates whether $(I_i, \{P'_1, \ldots, P'_t\})$ is feasible, where each P'_j is a path of ℓ_j vertices. (We allow ℓ_j to be nonpositive when P'_j is an empty graph with no vertex.) Clearly, $(I, \mathcal{P})$ is feasible if and only if $\mathsf{opt}(|I|; n_1, \ldots, n_t)$ is true. By the above observation, we have the following recurrence:

$$\mathsf{opt}(0; \ell_1, \ldots, \ell_t) = \begin{cases} \text{true} & \text{if } \ell_i \le 0 \text{ for all } 1 \le i \le t, \\ \text{false} & \text{otherwise,} \end{cases}$$

$$\mathsf{opt}(i; \ell_1, \ldots, \ell_t) = \bigvee_{1 \le j \le t} \mathsf{opt}(i - 1; \ell_1, \ldots, \ell_{j-1}, \ell_j - 2q_i - 1, \ell_{j+1}, \ldots, \ell_t).$$

We can evaluate this recurrence in time $n^{O(\ell^2)}$ since $\max_{1 \le i \le t} n_i \le n$ and $t \in O(\ell^2)$. This completes the proof. $\qquad\square$

5 Conclusion

We studied the parameterized complexity of GRAPH BURNING and presented an almost complete picture of the complexity hierarchy with respect to important structural parameters (see Fig. 1). One missing piece in the picture is the complexity of the problem parameterized by max leaf number. We showed that it is XP, but we do not know whether it is fixed-parameter tractable.

Another question is on approximation. It is known that GRAPH BURNING admits a 3-approximation algorithm [10] but it is APX-hard [41]. It would be natural to ask whether it admits a better approximation.

Finally, we recall that the following conjecture is still open: "$\mathsf{b}(G) \le \lceil \sqrt{n} \rceil$ for every connected n-vertex graph G."

References

1. Alon, N.: Transmitting in the *n*-dimensional cube. Discret. Appl. Math. **37**(38), 9–11 (1992). https://doi.org/10.1016/0166-218X(92)90121-P
2. Bessy, S., Bonato, A., Janssen, J.C.M., Rautenbach, D., Roshanbin, E.: Burning a graph is hard. Discret. Appl. Math. **232**, 73–87 (2017). https://doi.org/10.1016/j.dam.2017.07.016
3. Bessy, S., Bonato, A., Janssen, J.C.M., Rautenbach, D., Roshanbin, E.: Bounds on the burning number. Discret. Appl. Math. **235**, 16–22 (2018). https://doi.org/10.1016/j.dam.2017.09.012
4. Bodlaender, H.L., Thomassé, S., Yeo, A.: Kernel bounds for disjoint cycles and disjoint paths. Theor. Comput. Sci. **412**(35), 4570–4578 (2011). https://doi.org/10.1016/j.tcs.2011.04.039
5. Bonato, A.: A survey of graph burning. Contributions Discret. Math. **16**(1), 185–197 (2021). https://doi.org/10.11575/cdm.v16i1.71194
6. Bonato, A., English, S., Kay, B., Moghbel, D.: Improved bounds for burning fence graphs. Graphs Comb. **37**(6), 2761–2773 (2021). https://doi.org/10.1007/s00373-021-02390-x
7. Bonato, A., Gunderson, K., Shaw, A.: Burning the plane. Graphs Comb. **36**, 1311–1335 (2020). https://doi.org/10.1007/s00373-020-02182-9
8. Bonato, A., Janssen, J.C.M., Roshanbin, E.: Burning a graph as a model of social contagion. In: WAW 2014, vol 8882 of Lecture Notes in Computer Science, pp. 13–22, (2014). https://doi.org/10.1007/978-3-319-13123-8_2
9. Bonato, A., Janssen, J.C.M., Roshanbin, E.: How to burn a graph. Internet Math. **12**(1–2), 85–100 (2016). https://doi.org/10.1080/15427951.2015.1103339
10. Bonato, A., Kamali, S.: Approximation algorithms for graph burning. In: TAMC 2019, volume 11436 of Lecture Notes in Computer Science, pp. 74–92, (2019). https://doi.org/10.1007/978-3-030-14812-6_6
11. Bonato, A., Lidbetter, T.: Bounds on the burning numbers of spiders and path-forests. Theor. Comput. Sci. **794**, 12–19 (2019). https://doi.org/10.1016/j.tcs.2018.05.035
12. Corneil, D.G., Rotics, U.: On the relationship between clique-width and treewidth. SIAM J. Comput. **34**(4), 825–847 (2005). https://doi.org/10.1137/S0097539701385351
13. Courcelle, B., Makowsky, J.A., Rotics, U.: Linear time solvable optimization problems on graphs of bounded clique-width. Theory Comput. Syst. **33**(2), 125–150 (2000). https://doi.org/10.1007/s002249910009
14. Cygan, M., Fomin, F.V., Kowalik, Ł, Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: Parameterized Algorithms. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21275-3
15. Das, S., Dev, S.R., Sadhukhan, A., Sahoo, U.K., Sen, S.: Burning spiders. In: CALDAM 2018, volume 10743 of Lecture Notes in Computer Science, pp. 155–163, (2018). https://doi.org/10.1007/978-3-319-74180-2_13
16. Dom, M., Lokshtanov, D., Saurabh, S.: Kernelization lower bounds through colors and IDs. ACM Trans. Algorithms **11**(2), 13:1-13:20 (2014). https://doi.org/10.1145/2650261
17. Downey, R.G., Fellows, M.R.: Fixed-parameter tractability and completeness I: basic results. SIAM J. Comput. **24**(4), 873–921 (1995). https://doi.org/10.1137/S0097539792228228
18. Downey, R.G., Fellows, M.R.: Parameterized Complexity. Springer (1999). https://doi.org/10.1007/978-1-4612-0515-9
19. Fellows, M.R., Langston, M.A.: On well-partial-order theory and its application to combinatorial problems of VLSI design. SIAM J. Discret. Math. **5**(1), 117–126 (1992). https://doi.org/10.1137/0405010
20. Fitzpatrick, S.L., Wilm, L.: Burning circulant graphs. *CoRR*, (2017). arXiv:1706.03106
21. Foldes, S., Hammer, P.L.: Split graphs. In: the Eighth Southeastern Conference on Combinatorics, Graph Theory and Computing, volume 19 of Congressus Numerantium, pp. 311–315 (1977)
22. Fomin, F.V., Kratsch, D., Woeginger, G.J.: Exact (exponential) algorithms for the dominating set problem. In: WG 2004, volume 3353 of Lecture Notes in Computer Science, pp. 245–256, (2004) https://doi.org/10.1007/978-3-540-30559-0_21
23. Gajarský J., Lampis, M., Ordyniak, S.: Parameterized algorithms for modular-width. In: IPEC 2013, volume 8246 of Lecture Notes in Computer Science, pp. 163–176, (2013). https://doi.org/10.1007/978-3-319-03898-8_15
24. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, W. H (1979)

25. Grohe, M., Kreutzer, S., Siebertz, S.: Deciding first-order properties of nowhere dense graphs. J. ACM **64**(3), 17:1-17:32 (2017). https://doi.org/10.1145/3051095
26. Gurski, F.: The behavior of clique-width under graph operations and graph transformations. Theory Comput. Syst. **60**(2), 346–376 (2017). https://doi.org/10.1007/s00224-016-9685-1
27. Hiller, M., Triesch, E., Koster, A.M.C.A.: On the burning number of *p*-caterpillars. In: CTW 2020, volume 5 of AIRO Springer Series, pp. 145–156, (2020). https://doi.org/10.1007/978-3-030-63072-0_12
28. Ho, T.-Y., Hsu, L.-H., Sung, T.-Y.: Transmitting on various network topologies. Networks **27**(2), 145–157 (1996). https://doi.org/10.1002/(SICI)1097-0037(199603)27:2<145::AID-NET6>3.0.CO;2-K
29. Janssen, R.: The burning number of directed graphs: Bounds and computational complexity. Theory Appl. Graphs **7**(1), Article 8, (2020). https://doi.org/10.20429/tag.2020.070108
30. Jwo, J.-S., Tuan, T.-C.: On transmitting delay in a distance-transitive strongly antipodal graph. Inf. Process. Lett. **51**(5), 233–235 (1994). https://doi.org/10.1016/0020-0190(94)00105-7
31. Kamali, S., Miller, A., Zhang, K: Burning two worlds. In: SOFSEM 2020, volume 12011, pp. 113–124. Springer, (2020). https://doi.org/10.1007/978-3-030-38919-2_10
32. Kare, A.S., Vinod Reddy, I.: Parameterized algorithms for graph burning problem. In: IWOCA 2019, volume 11638 of Lecture Notes in Computer Science, pp. 304–314, (2019). https://doi.org/10.1007/978-3-030-25005-8_25
33. Kleitman, D.J., West, D.B.: Spanning trees with many leaves. SIAM J. Discret. Math. **4**(1), 99–106 (1991). https://doi.org/10.1137/0404010
34. Land, M.R., Lu, L.: An upper bound on the burning number of graphs. In: WAW 2016, volume 10088 of Lecture Notes in Computer Science. pp. 1–8, (2016). https://doi.org/10.1007/978-3-319-49787-7_1
35. Liu, H., Hu, X., Hu, X.: Burning number of caterpillars. Discret. Appl. Math. **284**, 332–340 (2020). https://doi.org/10.1016/j.dam.2020.03.062
36. Liu, H., Hu, X., Hu, X.: Burning numbers of path forests and spiders. Bull. Malays. Math. Sci. Soc. **44**, 661–681 (2021). https://doi.org/10.1007/s40840-020-00969-w
37. Liu, H., Zhang, R., Hu, X.: Burning number of theta graphs. Appl. Math. Comput. **361**, 246–257 (2019). https://doi.org/10.1016/j.amc.2019.05.031
38. Liu, Z., Sung, T.-Y.: Routing and transmitting problems in de Bruijn networks. IEEE Trans. Comput. **45**(9), 1056–1062 (1996). https://doi.org/10.1109/12.537129
39. Mitsche, D., Pralat, P., Roshanbin, E.: Burning graphs: a probabilistic perspective. Graphs Comb. **33**(2), 449–471 (2017). https://doi.org/10.1007/s00373-017-1768-5
40. Mitsche, Dieter, Pralat, Pawel, Roshanbin, Elham: Burning number of graph products. Theor. Comput. Sci. **746**, 124–135 (2018). https://doi.org/10.1016/j.tcs.2018.06.036
41. Mondal, D., Parthiban, N., Kavitha, V., Rajasingh, I.: APX-hardness and approximation for the *k*-burning number problem. In: WALCOM 2021, volume 12635 of Lecture Notes in Computer Science. pp. 272–283, (2021). https://doi.org/10.1007/978-3-030-68211-8_22
42. Nešetřil, J., de Mendez, P.O.: Sparsity: Graphs, Structures, and Algorithms. Algorithms and Combinatorics. Springer, Berlin (2012). https://doi.org/10.1007/978-3-642-27875-4
43. Oum, S.-I.: Approximating rank-width and clique-width quickly. ACM Trans. Algorithms **5**(1), 101–1020 (2008). https://doi.org/10.1145/1435375.1435385
44. Sorge, M., Weller, M.: The graph parameter hierarchy, 2019. URL: https://manyu.pro/assets/parameter-hierarchy.pdf
45. Tan, T.S., Teh, W.C.: Graph burning: tight bounds on the burning numbers of path forests and spiders. Appl. Math. Comput. (2020). https://doi.org/10.1016/j.amc.2020.125447