Abstract. The Firefighter problem is to place firefighters on the vertices of a graph to prevent a fire with known starting point from lighting up the entire graph. In each time step, a firefighter may be permanently placed on an unburned vertex and the fire spreads to its neighborhood in the graph in so far no firefighters are protecting those vertices. The goal is to let as few vertices burn as possible. This problem is known to be NP-complete, even when restricted to bipartite graphs or to trees of maximum degree three. Initial study showed the Firefighter problem to be fixed-parameter tractable on trees in various parameterizations. We complete these results by showing that the problem is in FPT on general graphs when parameterized by the number of burned vertices, but has no polynomial kernel on trees, resolving an open problem. Conversely, we show that the problem is \W[1]-hard when parameterized by the number of unburned vertices, even on bipartite graphs. For both parameterizations, we additionally give refined algorithms on trees, improving on the running times of the known algorithms.

1 Introduction

The Firefighter problem concerns a deterministic model of fire spreading through a graph via its edges. The problem has recently received considerable attention [9,14]. In the model, we are given a graph $G$ with a vertex $s \in V(G)$. At time $t = 0$, the fire breaks out at $s$ and vertex $s$ starts burning. At each step $t \geq 1$, first the firefighter protects one vertex not yet on fire—this vertex remains permanently protected—and the fire then spreads from burning vertices to all unprotected neighbors of these vertices. The process stops when the fire cannot spread anymore. The goal is to find a strategy for the firefighter that minimizes the amount of burned vertices, or, equivalently, maximizes the number of saved, i.e. not burned, vertices.

It is known that the Firefighter problem is NP-hard, even when restricted to bipartite graphs [14] or trees of maximum degree three [10]. However, it is polynomial-time solvable on such trees if the root has degree two [14]. We refer to the survey [11] for an overview of further combinatorial results on the problem. The study of the problem from the perspective of parameterized complexity...
was initiated by Cai, Verbin, and Yang [6]. They considered the following parameterized versions of the problem and obtained a number of parameterized algorithms on trees.

The first parameterization considered by Cai et al. in [6] is by the number of saved vertices.

**Saving $k$ Vertices**

**Parameter:** $k$

**Input:** An undirected graph $G$, a vertex $s$, and an integer $k$.

**Question:** Is there a strategy to save at least $k$ vertices when a fire breaks out at $s$?

Cai et al. proved that **Saving $k$ Vertices** on trees has a kernel with $O(k^2)$ vertices. They also gave a randomized algorithm solving **Saving $k$ Vertices** on trees in time $O(4^k + n)$, which can be derandomized to a $O(n + 2^{O(k)})$-time algorithm.

The second parameterization is by the number of burned vertices.

**Saving All But $k$ Vertices**

**Parameter:** $k$

**Input:** An undirected $n$-vertex graph $G$, a vertex $s$, and an integer $k$.

**Question:** Is there a strategy to save at least $n - k$ vertices when a fire breaks out at $s$?

For **Saving All But $k$ Vertices** on trees, Cai et al. gave a randomized algorithm of running time $O(4^k n)$, which can be derandomized to a $O(2^{O(k)} n \log n)$-time algorithm. They left as an open problem whether **Saving All But $k$ Vertices** has a polynomial kernel on trees.

The last parameterization is by the number of protected vertices, i.e. the number of vertices occupied by firefighters.

**Maximum $k$-Vertex Protection**

**Parameter:** $k$

**Input:** An undirected graph $G$, a vertex $s$, and an integer $k$.

**Question:** What is a strategy that saves the maximum number of vertices by protecting $k$ vertices when a fire breaks out at $s$?

For **Maximum $k$-Vertex Protection** on trees, Cai et al. gave a randomized algorithm of running time $O(k^{O(k)} n)$, which can be derandomized to a $O(k^{O(k)} n \log n)$-time algorithm. They left open whether the problem has a polynomial kernel on trees, and asked whether there is an algorithm solving the problem on trees in time $2^{o(k \log k)} n^{O(1)}$.

We will sometimes consider the decision variant of **Maximum $k$-Vertex Protection**.

**$k$-Vertex Protection**

**Parameter:** $k$

**Input:** An undirected graph $G$, a vertex $s$, an integer $k$, and an integer $K$.

**Question:** Is there a strategy that saves at least $K$ vertices by protecting $k$ vertices when a fire breaks out at $s$?

The unparameterized version of this problem is obviously NP-hard on trees of maximum degree three from the hardness of the **Firefighter** problem.

**Our Results.** We resolve several open questions of Cai, Verbin, and Yang [6]. We also refine and extend some of the results of [6].