Available online at www.sciencedirect.com
ScienceDirect
Discrete Mathematics 307 (2007) 2094–2105
DISCRETE MATHEMATICS
www.elsevier.com/locate/disc

The firefighter problem for graphs of maximum degree three☆

Stephen Finbow[a], Andrew King[a], Gary MacGillivray[a], Romeo Rizzi[b]

[a] *Department of Mathematics and Statistics, University of Victoria, Victoria, BC, Canada*
[b] *Department of Telecommunications and Informatics, University of Trento, Trento, Italy*

Received 8 September 2003; received in revised form 17 November 2004; accepted 12 December 2005
Available online 6 December 2006

Abstract

We show that the firefighter problem is NP-complete for trees of maximum degree three, but in P for graphs of maximum degree three if the fire breaks out at a vertex of degree at most two.

Keywords: Firefigher problem; NP-complete problem; Optimization on graphs

1. Introduction

We consider a dynamic problem introduced by Hartnell in 1995 [7]. Let (G, r) be a connected rooted graph. At time 0, a fire breaks out at r. At each subsequent time interval, the firefighter *defends* some vertex which is not on fire, and then the fire spreads to all undefended neighbours of each *burning* (i.e., on fire) vertex. Once burning or defended, a vertex remains so for all time intervals. The process ends when the fire can no longer spread. The firefighter optimization problem is to determine the maximum number of vertices that can be *saved*, i.e., that are not burning when the process ends. The firefighter decision problem is stated formally below:

FIREFIGHTER

INSTANCE: A rooted graph (G, r) and an integer $k \geqslant 1$.

QUESTION: If the fire breaks out at r, is there a strategy under which at most k vertices burn? That is, does there exist a finite sequence $d_1, d_2, \ldots, d_t$ of vertices of G such that, if the fire breaks out at r, then,

(i) vertex d_i is neither burning nor defended at time i,
(ii) at time t no undefended vertex is adjacent to a burning vertex, and
(iii) at most k vertices are burned at the end of time t.

Papers investigating the firefighter problem have appeared in the literature. Algorithms for two- and three-dimensional grid graphs are presented in [11]. These lead to bounds on the maximum number of vertices that can be saved.

☆ Research supported by The Natural Sciences and Engineering Council of Canada.
E-mail address: aking6@cs.mcgill.ca (A. King).

0012-365X/$ - see front matter
doi:10.1016/j.disc.2005.12.053

NP-completeness of the firefighter problem on bipartite graphs is established in [10]. This paper also establishes improved bounds and some exact values for the maximum number of vertices that can be saved for two-dimensional grids, and considers the restriction of the problem to trees. The results include exponential algorithms for solving the firefighter problem on trees (one of these runs in linear time for binary trees), and a polynomial-time algorithm for a subclass of trees related to perfect graphs. It is proved in [9] that the greedy algorithm is a 2-approximation algorithm on trees, that is, the maximum number of vertices saved is never more than twice the number saved using the greedy algorithm. (It need not be the case that the number of vertices burned under a greedy strategy is at most twice the number of vertices burned under an optimum strategy.) The firefighter problem on infinite grids is considered in [3]. The questions considered include the number of firefighters needed to contain the fire, or fires, and how the firefighters should proceed in the case where the fire burns for t time units before they arrive. Other aspects of the firefighter problem are studied in [2]. Related topics are examined in [1,5,6,8].

Perhaps the most interesting open question about the firefighter problem is its complexity for trees. A formal conjecture has never appeared in the literature, but it has been widely believed for some time that the problem is NP-complete for trees. We prove that this is indeed the case, and more. A sequence of transformations with increasing expressive power is used to show that the problem is NP-complete for trees of maximum degree three. By contrast, we show that if the fire breaks out at a vertex of degree two, the problem can be solved in polynomial time for graphs of maximum degree three.

2. Problems and preliminaries

In this section, we introduce the decision problems used in our sequence of reductions, and establish several preparatory lemmas.

3-T-FIRE

INSTANCE: A rooted tree (T, r) with maximum degree $\Delta(T) \leqslant 3$ and a positive integer k.

QUESTION: If the fire breaks out at r, is there a strategy such that at most k vertices burn?

3-T′-FIRE

INSTANCE: A rooted tree (T, r) such that $d(r) = 2^m + 2$ for some positive integer m and every other vertex in T has degree at most 3, and a positive integer k.

QUESTION: If the fire breaks out at r, is there a strategy such that at most k vertices burn?

The next problem is commonly known as Not All Equal 3-SAT without negated literals, or Hypergraph 2-Colourability [4].

NAE 3-SAT WITHOUT NEGATED LITERALS

INSTANCE: An ordered pair (B, C) consisting of a set B of boolean variables and a set C of clauses over B in conjunctive normal form, each containing three non-negated literals.

QUESTION: Is there a truth assignment for B such that every clause in C contains at least one true literal and at least one false literal?

For our purposes, it is helpful to work with a variation of the above problem.

RESTRICTED NAE 3-SAT

INSTANCE: An ordered pair (B, C) consisting of a set B of boolean variables and a set C of clauses over B in conjunctive normal form, where $|B| = 2^m$ for some integer $m \geqslant 2$, exactly $|C|/2$ clauses have no negated literals, and the remaining clauses are obtained from these by replacing each literal with its negation.

QUESTION: Is there a truth assignment for B such that every clause in C contains at least one true literal and at least one false literal?

Proposition 1. *RESTRICTED NAE 3-SAT is NP-complete.*

Proof. The transformation is from NAE 3-SAT without negated literals. Consider an instance (B, C) of NAE 3-SAT without negated literals. We construct an instance (B', C') of RESTRICTED NAE 3-SAT. Let $m = \lceil \log_2 |B| \rceil$. Construct B' by adding $2^m - |B|$ new boolean variables to B. The collection C' of clauses is formed from C by adding, for every $c \in C$, the clause formed by negating every literal in c. The transformation can clearly be accomplished in polynomial time.

Suppose B has a truth assignment such that each clause in C contains a true literal and a false literal. Then, since no variable in $B' - B$ appears in a clause in C', assigning each variable in $B' - B$ the value FALSE extends this truth assignment to one for B' with the property that each clause in C' contains a true literal and a false literal.

Conversely, suppose there is a truth assignment for B' such that each clause in C' contains a true literal and a false literal. Then, since $B \subseteq B'$ and $C \subseteq C'$, the restriction of this truth assignment to B has the property that each clause in C contains a true literal and a false literal. □

3. Trees

In this section we will establish NP-completeness of the firefighter problem for rooted trees of maximum degree three, and in which the root has degree three.

Since any two vertices of a tree are joined by a unique path, if some ancestor of a vertex v is defended, then v is saved. We shall also say that v is *protected*.

Our reductions will make use of two classes of graphs. We call a graph of the type shown in Fig. 1(i), and rooted at the vertex of degree two, a *snake*. We will denote a snake of diameter n with distance m from a to b by $\mathscr{S}(n, m)$. A *snake tree* is a spanning tree of a snake of the form shown in Fig. 1(ii). We call a graph of the type shown in Fig. 1(iii), and rooted at the vertex of degree two, a *ladder*. We use $\mathscr{L}(n)$ to denote a ladder of diameter n. A *ladder tree* is a spanning tree of a ladder of the form shown in Fig. 1(iv).

As our reductions are based on complicated arguments involving the number of vertices in certain subtrees, the following observations will be useful:

(i) The graph $\mathscr{S}(n, m)$ has $2n - 1$ vertices, of which $2m$ belong to the unique cycle containing a and b.
(ii) The graph $\mathscr{L}(n)$ has $2n + 1$ vertices.

A rooted tree (T, r) is called *full* if all leaves occur at the same level (i.e. all leaves are at the same distance from the root).

A binary tree (T, r) is called *complete* if every internal vertex has exactly two children. Thus, a complete and full binary tree of height h has exactly $2^{h+1} - 1$ vertices of which 2^h are leaves, each of which is at distance h from r.

Let x be a vertex of a graph G, and let (T, r) be a rooted graph. When we *root a copy of T at x*, we construct a new graph from the disjoint union of G and T by identifying the vertices x and r.

In the constructions that follow, we will normally root either complete and full binary trees, or paths, at vertices of other graphs. We always assume that the root vertex of a path is an endvertex.

Theorem 1. 3-T′-*FIRE is NP-complete*.

Proof. The transformation is from RESTRICTED NAE 3-SAT. Suppose an instance (B, C) of RESTRICTED NAE 3-SAT, where $B = \{b_1, b_2, \ldots, b_b\}$, the integer m is defined by $b = 2^{m-1}$, and $C = \{c_1, c_2, \ldots, c_n\}$, is given. Assume that $n > b \geqslant 4$, (clauses can be duplicated to ensure $n > b$; this assumption will simplify analysis). Also assume that for

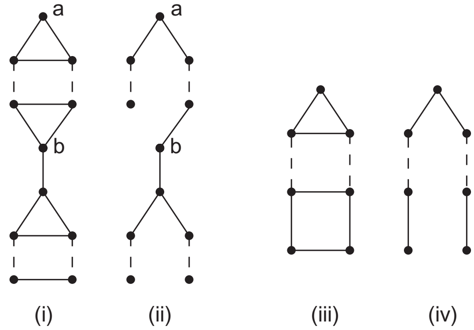

Fig. 1. (i) Snake; (ii) snake tree; (iii) ladder; (iv) ladder tree.

$i = 1, 2, \ldots, n/2$, the clause $c_{2(i+1)}$ arises from negating each variable in the clause c_{2i+1}. Let $p = \lceil \log_2(n) \rceil + 2$. The construction of the rooted tree in our instance of 3-T′-FIRE proceeds in two phases. First, we construct a full rooted tree (T_1, r) with height $b + p$ in which the degree of r is $2^m + 2$. We subsequently augment (T_1, r), without changing the degree of r, to construct our final rooted tree (T, r).

Starting with the single vertex r, proceed as follows. For $i = 1, 2, \ldots, b$, root two paths of length i at r and call the vertices of degree one in the resulting graph b_i and $\bar{b}_i$. At each of b_i and $\bar{b}_i$, root a complete and full binary tree of height p. From each leaf of these trees root a path of the appropriate length so that the vertex of degree one in the resulting graph is distance $b + p$ from r. (The paths rooted at b_b and $\bar{b}_b$ have length zero.) Call these leaves $t_{b_i,1}, t_{b_i,2}, \ldots, t_{b_i,2^p}$ and $t_{\bar{b}_i,1}, t_{\bar{b}_i,2}, \ldots, t_{\bar{b}_i,2^p}$, respectively. Next, root two paths of length $b + 1$ at r and call the resulting vertices of degree one b_0 and $\bar{b}_0$. From these vertices, root complete and full binary trees of height $p - 1$, calling their leaves $t_{b_0,1}, t_{b_0,2}, \ldots, t_{b_0,2^{p-1}}$ and $t_{\bar{b}_0,1}, t_{\bar{b}_0,2}, \ldots, t_{\bar{b}_0,2^{p-1}}$, respectively. The tree constructed so far is (T_1, r) (it will arise in the argument below). Note that $d(r) = 2^m + 2$.

The number of vertices of T_1 is

$$
\begin{aligned}
|V(T_1)| = {} & 1 + 2(1 + 2 + \cdots + b) + 2b(2^{p+1} - 2) \\
& + 2 \cdot 2^p((b-1) + (b-2) + \cdots + 0) + 2(b+1) + 2(2^p - 2)
\end{aligned}
$$

(the number of these that are leaves is $2b \cdot 2^p + 2 \cdot 2^{p-1} = (2b+1)2^p$). By definition of p we have $2^p < 8n$, hence this phase of the construction can be carried out in polynomial time.

For the second phase of the construction, form (T, r) by augmenting (T_1, r) as follows. For $1 \leqslant j \leqslant 2^{p-1}$, add children x_j and y_j from vertex $t_{b_0,j}$, and children $\bar{x}_j$ and $\bar{y}_j$ from $t_{\bar{b}_0,j}$. At each of the vertices just added, root a copy of $\mathscr{L}^T(3n+1)$. For each i and j with $1 \leqslant i \leqslant b$ and $1 \leqslant j \leqslant n$, do the following: if b_i is in clause c_j, root a copy of $\mathscr{S}^T(3n+2, 3j-1)$ at $t_{b_i,j}$ and a copy of $\mathscr{S}^T(3n+2, 3j)$ at $t_{\bar{b}_i,j}$. If $\bar{b}_i$ is in clause c_j, root a copy of $\mathscr{S}^T(3n+2, 3j-1)$ at $t_{\bar{b}_i,j}$ and a copy of $\mathscr{S}^T(3n+2, 3j)$ at $t_{b_i,j}$. At each remaining unaltered leaf of T_1, root a copy of $\mathscr{L}^T(3n+2)$. This completes the construction.

We now calculate the number of vertices of T. For $i = 1, 2, \ldots, b$, let n_i denote the number of clauses containing b_i. By definition of RESTRICTED NAE 3-SAT, there are also exactly n_i clauses containing $\bar{b}_i$. Each clause contains exactly three literals, thus

$$
\sum_{i=1}^{n} n_i = \frac{3n}{2}.
$$

Since each copy of $\mathscr{L}^T(3n+1)$ has $6n+3$ vertices, each copy of $\mathscr{L}^T(3n+2)$ has $6n+5$ vertices, and both $\mathscr{S}^T(3n+2, 3j-1)$ and $\mathscr{S}^T(3n+2, 3j)$ have $6n+3$ vertices,

$$
\begin{aligned}
|V(T)| &= |V(T_1)| + 2 \cdot 2^{p-1} \cdot 2 + 2 \cdot 2^{p-1} \cdot 2(6n + 3 - 1) \\
&\quad + 2\sum_{i=1}^{b} 2n_i(6n + 3 - 1) + 2\sum_{i=1}^{b} (2^p - 2n_1)(6n + 5 - 1) \\
&= |V(T_1)| + 2^p(6n+3) + 2b \cdot 2^p(6n+4) - 12b.
\end{aligned}
$$

Thus, this phase of the construction can also be carried out in polynomial time.

To complete the instance of 3-T′-FIRE, set

$$
k = |V(T)| - \left(\sum_{i=1}^{b} [2^p(6n + 6 + b - i) - 1] + \sum_{i=0}^{p} [2^{p-i}(6n+4) - 1] + 9n^2 + \frac{15n}{2} + 1 \right).
$$

The reasoning behind the value of k will become apparent in the remainder of the argument.

The height of (T, r) is $d = b + p + 3n + 2$. For $i = 1, 2, \ldots, d$, let w_i be the largest number of vertices in a subtree of (T, r) rooted at level i.

Claim 1. *The quantity*

$$w_i = \begin{cases} 2^p(6n+6+b-i)-4n_i-1, & 1\leqslant i\leqslant b, \\ 2^{p-i+b-1}(6n+4)-1, & b+1\leqslant i\leqslant b+p+1, \\ 2(d-i)+1, & b+p+2\leqslant i\leqslant d \\ & \quad \text{and } i-(b+p+2)\not\equiv 0\,(\mathrm{mod}\,3), \\ 2(d-i) & b+p+2\leqslant i\leqslant d \\ & \quad \text{and } i-(b+p+2)\equiv 0\,(\mathrm{mod}\,3). \end{cases}$$

The claim is proved by considering each case in turn.

Case 1: $1\leqslant i\leqslant b$. Let $v_i=2^p(6n+6+b-i)-4n_i-1$. We shall show that the subtree rooted at b_i, or $\bar{b}_i$, has v_i vertices, and the subtree rooted at any other vertex at level i has at most v_i-n vertices.

The subtree rooted at b_i, or $\bar{b}_i$, consists of a full, complete, binary tree of height p, a path from each of its leaves to depth $b+p$, $2n_i$ snake trees, and 2^p-2n_i ladder trees. The number of vertices in this subtree is therefore

$$(2^{p+1}-1)+2^p(b-i)+2^p(6n+4)-4n_i=v_i.$$

Similar counting shows that for $0<j\leqslant b+1-i<n$ the number of vertices in the subtree rooted at the ancestor of b_{i+j}, or $\bar{b}_{i+j}$, at level i is

$$\begin{aligned} j+2^p(6n+6+b-i-j)-4n_{i+j} &= v_i+4n_i-4n_{i+j}-2^pj+j+1 \\ &\leqslant v_i+4n_i-4n_{i+j}-4n+j+1 \\ &\leqslant v_i-4n_{i+j}-2n+j+1 \\ &\leqslant v_i-n, \end{aligned}$$

where we have used the fact that, by definition of RESTRICTED NAE 3-SAT, $n_i\leqslant n/2$. Similarly again, for $1\leqslant j\leqslant p$ and $i>j$ the number of vertices in a subtree rooted at a descendant of b_{i-j}, or $\bar{b}_{i-j}$, at level i is at most

$$\begin{aligned} 2^{p-j}(6n+6+b-i+j)-1 &\leqslant v_i-2^pj-(2^p-2^{p-j})(6n+6+b-i)+4n_i \\ &\leqslant v_i+2^pj-2^{p-1}(6n+6+b-i)+2n \\ &\leqslant v_i+8nj-12n^2-12n-2nb+2ni+2n \\ &\leqslant v_i+8n^2-12n^2-12n-2nb+2n^2+2n \\ &\leqslant v_i-n, \end{aligned}$$

where we have used the inequality $j<i\leqslant b<n$. For $i>j>p$, the number of vertices in the subtree rooted at a descendant of b_{i-j} or $\bar{b}_{i-j}$ is at most

$$6n+3+(b+p)-i\leqslant v_i-n$$

Finally, consider the subtree rooted at the ancestor of b_0 or $\bar{b}_0$ at level i. Counting similarly to the above, it contains

$$\begin{aligned} (b-i+1)+(2^p-1)+2\cdot 2^{p-1}(6n-3) &\leqslant v_i-2\cdot 2^p-2^p(b-i)+4n_i+1+(b-i+1) \\ &\leqslant v_i-8n+4n_i+1 \\ &\leqslant v_i-n \end{aligned}$$

vertices. This completes the proof of Case 1.

Case 2: $b+1 \leqslant i \leqslant b+p+1$. In this case, set

$$v_i = 2^{p-(i-b)+2} - 1 + 2^{p-(i-b)+1}(6n+2) = 2^{p-(i-b)+1}(6n+4) - 1.$$

Suppose first that $b+1 \leqslant i \leqslant b+p$. Consider a subtree rooted at a descendant of b_0 or $\bar{b}_0$ at level i (all such subtrees are isomorphic). It contains a full, complete binary tree of height $p-(i-b)+1$ with $2^{p-(i-b)+1}$ copies of $\mathscr{L}^T(3n+1)$ rooted at its leaves. Thus, it has

$$(2^{p-i+b+2} - 1) + 2^{p-(i-b)+1}(6n+3-1) = v_i$$

vertices. Now consider a subtree rooted at a descendant of $b_1, \bar{b}_1, b_2, \bar{b}_2, \ldots, b_b, \bar{b}_b$ at level i. It has at most $2^{p-(i-b)+1} - 1$ vertices at levels i through $b+p$ and no more than $2^{p-(i-b)}$ vertices at level $b+p$. The maximum number of vertices in the subtree rooted at one of these is $6n+5$. Therefore, the number of vertices is at most

$$\begin{aligned} 2^{p-(i-b)+1} - 1 + 2^{p-(i-b)}(6n+4) &\leqslant 2^{p-(i-b)}(6n+6) - 1 \\ &= 2^{p-(i-b)+1}(3n+3) - 1 \\ &\leqslant v_i - 2^{p-(i-b)+1}(3n+1) \\ &\leqslant v_i - n. \end{aligned}$$

Finally, consider the case $i = b+p+1$. The maximum number of vertices in the subtree rooted at a descendant of $b_1, \bar{b}_1, b_2, \bar{b}_2, \ldots, b_b, \bar{b}_b$ at level i is $6n+1 < 6n+3 = v_i$ (as the snake trees and ladder trees are rooted at level $b+p$). This completes the proof of Case 2.

Case 3: $b+p+2 \leqslant i \leqslant d = b+p+3n+2$. A subtree rooted at level i is a subtree of a snake tree or a ladder tree which is rooted at level $b+p$ of T. Thus, for $\ell = 0, 1, \ldots, n$, there is no vertex with two children at level $b+p+3\ell+2$ of T. By definition of the snake trees $\mathscr{S}^T(3n+2, 3j)$ and $\mathscr{S}^T(3n+2, 3j-1)$, we then have

$$w_i = \begin{cases} 2(d-i)+1, & b+p+2 \leqslant i \leqslant d \quad \text{and } i-(b+p+2) \not\equiv 0 \pmod 3, \\ 2(d-i), & b+p+2 \leqslant i \leqslant d \quad \text{and } i-(b+p+2) \equiv 0 \pmod 3. \end{cases}$$

This completes the proof of the claim. $\square$

We must show that the answer for our instance of 3-T'-FIRE is YES if and only if the answer for the given instance of RESTRICTED NAE 3-SAT is YES. We will do so by arguing that if (B, C) has a satisfying truth assignment then there is a strategy for the firefighter problem on (T, r) under which at most k vertices are burned, and if (B, C) has no satisfying truth assignment then no such strategy exists.

It is proved in [10] that an optimal strategy for the firefighter problem on a tree defends, for $i \geqslant 1$, a vertex on level i at time i until the fire can no longer spread. Therefore, in what follows we consider only strategies of this type.

Let σ be a strategy for the firefighter problem on (T, r). For $i \geqslant 1$, we let κ_i^σ be the number of vertices in the subtree whose root is defended at level i under σ.

Let τ be a truth assignment for the variables in B. We define the *truth assignment strategy* $f(\tau)$ for the firefighter problem as follows: For $i = 1, 2, \ldots, b$, if b_i is true, defend b_i at time i and otherwise defend $\bar{b}_i$ at time i. At time $b+1$, defend $\bar{b}_0$. From time $b+2$ to $b+p$, defend the unprotected descendant of b_0 which is not on the path from r to x_1. At time $b+p+1$, defend x_1. For time $i = b+p+2$ to $b+p+3n+2$, defend the tree greedily, that is, at time i defend a vertex at level i with the largest number of descendants. Note that, in case of a tie, the subtrees rooted at each such vertex are isomorphic. Thus, assuming a predetermined tie-breaking scheme, the function $f(\tau)$ is well defined.

Claim 2. *If τ is a satisfying truth assignment for (B, C), then the number of vertices saved by $f(\tau)$ is $\sum_{i=1}^{d} \kappa_i^{f(\tau)} = |V(T)| - k$.*

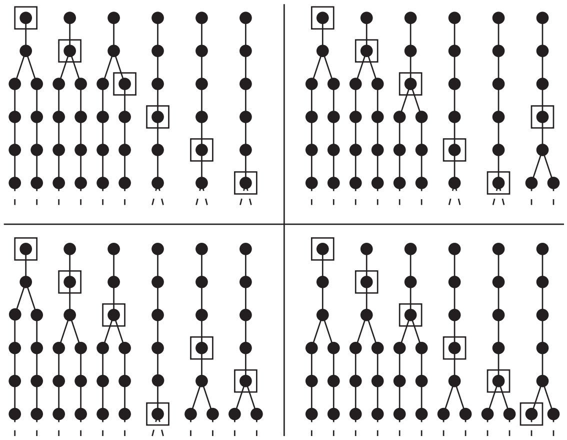

Fig. 2. The possible configurations for snake trees in an clause pair unprotected as of time $b+p+1$. Clockwise from top left, these correspond to the cases where the first clause has exactly 3, 2, 0, and 1 true literals.

Proof. For $1 \leqslant i \leqslant b+p+1$ it follows from the definition of $f(\tau)$ and our earlier argument that $\kappa_i^{f(\tau)} = w_i$. We must determine $\kappa_i^{f(\tau)}$ for $i \geqslant b+p+2$.

For $1 \leqslant j \leqslant n/2$, consider the *clause pair* c_{2j-1}, c_{2j}. In the construction of (T, r), these two clauses gave rise to 12 snake trees:

(i) For each of the three literals b in c_{2j-1} there is a copy of $\mathscr{S}^T(3n+2, 3(2j-1)-1)$ rooted at $t_{b,2j-1}$, and a copy of $\mathscr{S}^T(3n+2, 3(2j-1))$ rooted at $t_{\bar{b},2j-1}$;

(ii) For each of the three literals $\bar{b}$ in c_{2j} there is a copy of $\mathscr{S}^T(3n+2, 3(2j)-1)$ rooted at $t_{\bar{b},2j}$, and a copy of $\mathscr{S}^T(3n+2, 3(2j))$ rooted at $t_{b,2j}$.

Since τ is a satisfying truth assignment, the definition of RESTRICTED NAE 3-SAT implies that one of c_{2j-1}, and c_{2j} has exactly one true literal while the other has exactly two true literals. Suppose c_{2j-1} has exactly two true literals. Then, among the 12 snake trees mentioned above, one copy of $\mathscr{S}^T(3n+2, 3(2j-1)-1)$, two copies of $\mathscr{S}^T(3n+2, 3(2j-1))$, two copies of $\mathscr{S}^T(3n+2, 3(2j)-1)$, and one copy of $\mathscr{S}^T(3n+2, 3(2j))$ have no vertex defended at time $b+p+2$. The case where c_{2j-1} has exactly one true literal is similar.

Thus, for $j=1, 2, \ldots, n/2$ and $\ell=0, 1, \ldots, 5$ the strategy $f(\tau)$ defends vertices at level $i=b+p+2+6(j-1)+\ell$ as shown in the upper right and lower left parts of Fig. 2, depending on which member of the clause pair has exactly one true literal. Note that exactly $3n/2$ of these defended vertices have exactly one child.

Since one vertex can be saved at level d as well, it follows that

$$\begin{aligned}\sum_{i=b+p+2}^{d} \kappa_i^{f(\tau)} &= \sum_{i=b+p+2}^{d} [2(d-i)+1] - \frac{3n}{2} \\ &= \sum_{i=0}^{3n} [2i+1] - \frac{3n}{2} \\ &= 9n^2 + \frac{9n}{2} + 1.\end{aligned}$$

The number of vertices saved by $f(\tau)$ is then

$$\begin{aligned}\sum_{i=1}^{d} \kappa_i^{f(\tau)} &= \sum_{i=1}^{b} [2^p(6n+6+b-i) - 2n_i - 1] + \sum_{i=0}^{p} [2^{p-i}(6n+4) - 1] + 9n^2 + \frac{9n}{2} + 1 \\ &= \sum_{i=1}^{b} [2^p(6n+6+b-i) - 1] + \sum_{i=0}^{p} [2^{p-i}(6n+4) - 1] + 9n^2 + \frac{15n}{2} + 1 \\ &= |V(T)| - k.\end{aligned}$$

This completes the proof of the claim. □

We now prove the converse of Claim 2, that if there is no satisfying truth assignment for (B, C) then every strategy results in more than k vertices being burned, in several steps.

Define the integer $\beta = 9n^2 + \frac{21}{2}n + 4$.

Claim 3. *Suppose (B, C) has no satisfying truth assignment. Then, for any strategy σ, if $\sum_{i=b+p+1}^{d} \kappa_i^\sigma \geqslant \beta$ then there is no truth assignment strategy which is the same as σ on every level from 1 to $b + p$.*

Suppose to the contrary that there exists such a strategy σ and a truth assignment τ such that σ is the same as τ on every levels from 1 to $b+p$. Then, at time $b+p$, there are $3n$ snake trees whose root is burning (as well as some ladder trees).

At most w_{b+p+1} vertices can be saved by any vertex defended at level $b + p + 1$ under σ. Since $\kappa_d^\sigma = 1$, it remains to consider the number of vertices that can be saved under σ at times $i = b + p + 1, b + p + 2, \ldots, b + p + 3n + 1$. It is easy to see that defending a vertex in a snake tree saves at least as many vertices as defending a vertex in a ladder tree, so we assume σ defends in snake trees at all of these times.

It follows from the definition of T that, for $1 \leqslant j \leqslant n/2$, at most 33 vertices can be protected on levels $b+p+2+6(j-1)$ through $b + p + 1 + 6j$ by the vertices defended at times $b + p + 2 + 6(j - 1)$ through $b + p + 1 + 6j$ (see Fig. 2). Further, 33 vertices can be saved if and only if both elements of the clause pair c_{2j-1}, c_{2j} are satisfied by τ. Below level $b + p + 1 + 6j$, no more than $36(n - 2j) + 12$ vertices can be saved by the vertices defended on these six levels. Since (B, C) has no satisfying truth assignment, there is a clause pair in which at least one member is not satisfied by τ. Thus, at levels $b + p + 1$ through d, the number of vertices saved must be fewer than

$$\begin{aligned}w_{b+p+1} + \sum_{j=1}^{n/2} [36(n-2j) + 45] + 1 &= w_{b+p+1} + \frac{45n}{2} + 36 \sum_{j=1}^{n/2} [n - 2j] + 1 \\ &= w_{b+p+1} + \frac{45n}{2} + 36\left(\left(\frac{n}{2}\right)^2 - \frac{n}{2}\right) + 1 \\ &= 9n^2 + \frac{21n}{2} + 4 \\ &= \beta.\end{aligned}$$

This proves the claim. □

Claim 4. *Suppose a strategy σ differs from all truth assignment strategies on level i for some i with $1 \leqslant i \leqslant b + p$. Then $\kappa_i^\sigma < w_i - n/2$.*

This follows immediately from our work when calculating w_i.

By Claim 1,

$$\sum_{i=b+p+1}^{d} w_i = (6n+3) + \left[\sum_{i=b+p+2}^{d} [2(d-i)+1]\right] - n$$

$$= 5n + 3 + \sum_{\ell=0}^{3n} [2\ell + 1]$$

$$= 9n^2 + 10n + 4$$

$$= \beta + \frac{n}{2}.$$

Thus, for any strategy σ,

$$\sum_{i=b+p+1}^{d} \kappa_i^{\sigma} \leqslant \beta + \frac{n}{2}.$$

Together, Claims 3 and 4 imply that if there is no satisfying truth assignment for (B, C), then for any strategy σ,

$$\sum_{i=1}^{d} \kappa_i^{\sigma} < \beta + \sum_{i=1}^{b+p} w_i = |V(T)| - k.$$

This completes the proof. $\square$

Theorem 2. *3-T-FIRE is NP-complete.*

Proof. The transformation is from RESTRICTED NAE 3-SAT. Given an instance (B, C) of RESTRICTED NAE 3-SAT, first construct (T, r) and k as in Theorem 1. In what follows we continue to use the notation from that proof. We will use (T, r) and k to obtain a rooted tree (T', r') with maximum degree three (and $\deg(r') = 3$) and an integer k' so that at most k vertices burn in (T, r) if and only if at most k' vertices burn in (T', r').

The construction of T' begins with the single vertex r'. Join three new vertices to r' and, at each of these, root a full, complete binary tree of height $m-1$ (where m is defined by $b = 2^{m-1}$, as in Theorem 1). For the moment consider the tree constructed so far as being ordered, so that its leaves are ordered from left to right. At each of the first $2^m - 1$ of these leaves, root a copy of a full, complete binary tree F of height $h = \lceil \log_2 |V(T)| \rceil + 3$. Label the remaining leaves from left to right as $r_0, r_1, \ldots, r_b$.

For $i = 0, 1, \ldots, b$, let (R_i, w_i) and $(\bar{R}_i, \bar{w}_i)$ denote the subtree of T rooted at the unique neighbour of r on the (r, b_i)-path and $(r, \bar{b}_i)$-path, respectively. Let (S_i, x_i) be the rooted tree constructed from (R_i, w_i) and $(\bar{R}_i, \bar{w}_i)$ by adding a new vertex x_i and joining it to w_i and $\bar{w}_i$.

To complete the construction of T', for $i = 1, 2, \ldots, b$, root a copy of S_i at r_i (see Fig. 3). Finally, set $k' = k + 2^m - 1 + m$.

We claim the construction can be carried out in polynomial time. The tree T' has $1 + 3(2^m - 1) \leqslant 3|V(T)|$ vertices on levels 0 through m. To estimate the number of vertices at levels greater than m, note that every copy of F has at most $32|V(T)|$ vertices, and there are fewer than $2(2b+2)$ such copies. Therefore, T' has fewer than $64(2b+2)|V(T)| + |V(T)|$ vertices on levels greater than or equal to $m+1$. Hence, the number of vertices of T' is polynomial in the size of the instance of RESTRICTED NAE 3-SAT. Since all steps in the construction can be carried out in polynomial time, the claim is proved.

Recall that in an optimal strategy for the firefighter problem on a tree, the vertex defended at time ℓ must be at level ℓ. By construction of T', the number of vertices on levels 0 through m that burn under any optimal strategy is $1 + m + 2^m - 1 = 2^m + m$. If no vertex among $r_0, r_1, \ldots, r_b$ is defended, then from time m onwards the firefighter problem on T' is (essentially) the same as the firefighter problem on T: this is because identifying the parents of $r_0, r_1, \ldots, r_b$ would yield a copy of T rooted at this new vertex. Thus, it suffices to show that if one of $r_0, r_1, \ldots, r_b$ is protected then more than k' vertices of T' burn.

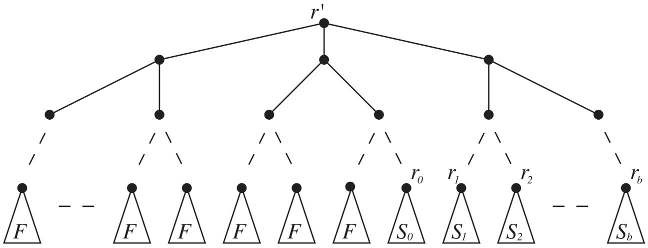

Fig. 3. The construction of T' from T.

We now establish a useful observation. Consider a full, complete binary tree K of height α, and suppose the root and both of its neighbours are burning. Then, under any strategy, at least $1 + (2^\alpha - 1) + \alpha = 2^\alpha + \alpha \geqslant (|V(K)| + 1)/2$ vertices of K burn.

Suppose first that one of $r_0, r_1, \ldots, r_b$ has a defended ancestor. Then there are two copies of F whose root eventually burns, and thus a copy of F for which the root and both of its neighbours are burning. By the above observation, more than half of the vertices of this copy of F-at least $8|V(T)|$ vertices—burn. Since $8|V(T)| \geqslant |V(T) \geqslant k + 2n \geqslant k + 2^m + m - 1 = k'$, Thus, if one of $r_0, r_1, \ldots, r_b$ has a defended ancestor then more than k' vertices of T' burn.

It remains to consider the case where none of $r_0, r_1, \ldots, r_b$ has a defended ancestor, but one of them is defended at time m. In this case, the root of some copy of F is burning. By the above observation and the definition of F, vertices belonging to this copy of F must be defended at times $m + 1, m + 2$ and $m + 3$, otherwise more than k' vertices will burn. This means that among the vertices corresponding to $T - r$, we cannot save more than (in the notation of the previous reduction)

$$
\begin{aligned}
\left(\sum_{i=1}^{d} w_i\right) + w_1 - w_2 - w_3 &\leqslant \left(\sum_{i=1}^{b+p} w_i\right) + \beta + \frac{n}{2} + w_1 - w_2 - w_3 \\
&\leqslant \left(\sum_{i=1}^{b+p} w_i\right) + \beta + \frac{n}{2} - 2^p(6n + 6 + b - 4) + 2n - 1 \\
&\leqslant \left(\sum_{i=1}^{b+p} w_i\right) + \beta + \frac{n}{2} - 2^p(6n + b) \\
&< \left(\sum_{i=1}^{b+p} w_i\right) + \beta \\
&\leqslant |V(T)| - k.
\end{aligned}
$$

That is, at least $k + 1$ vertices in this part of the tree burn. Since at least $2^m + m$ vertices on levels 0 through m burn under any strategy, at least $(k + 1) + 2^m + m > k'$ vertices of T' burn. This completes the proof. □

4. Graphs with maximum degree three rooted at a vertex of degree two

The results of the last section imply that FIREFIGHTER is NP-complete for rooted graphs (G, r) of maximum degree three and such that $\deg(r) = 3$. In this section we show that the firefighter problem is polynomially solvable for rooted graphs (G, r) of maximum degree three and such that $\deg(r) \leqslant 2$. If $\deg(r) = 1$ then the game is over after one move, so we assume in what follows that $\deg(r) = 2$.

We first define three sets that are used in the algorithm. Let V_1 be the set of vertices of degree one in G, and let V_2 be the set of all vertices of degree two in G. Define V_c as the set of vertices which belong to a cycle in G. For a vertex $u \in V_c$, let $C(u)$ denote the length of a shortest cycle containing u.

Next, we define a function $f : (V_1 \cup V_2 \cup V_c) \to \mathbb{Z}^+$:

$$f(u) = \begin{cases} \operatorname{dist}(u, r) + 1 & \text{if } u \in V_1 \cup V_2, \\ \operatorname{dist}(u, r) + C(u) - 1 & \text{if } u \in V_c \backslash V_2. \end{cases}$$

Strategy 1. *Begin by finding some $u \in (V_1 \cup V_2 \cup V_c)$ such that $f(u) = \min\{f(x) \mid x \in (V_1 \cup V_2 \cup V_c)\}$.*

Case 1: If $u \in V_1 \cup V_2$, then find a shortest path P from v to u. At each turn, defend the vertex which is adjacent to a burning vertex but neither burning itself nor on P. If $u \in V_2$, then at turn $f(u)$ defend the neighbour of u which is not on burning. (Clearly $f(u)$ vertices are burned.)

Case 2: Suppose $u \in V_c \backslash V_2$. Let C be the shortest cycle containing u, and let P be a shortest path from v to u. At each turn from 1 to $\operatorname{dist}(v, u)$, defend the vertex adjacent to a burning vertex but neither burning itself nor on P. On turn $\operatorname{dist}(v, u) + 1$, defend either unburned vertex on C with a burning neighbour. On each subsequent turn, defend the vertex not on C that has a burning neighbour. (A total of $f(u)$ vertices are burned: each vertex in P, and each vertex on C except one.)

Lemma 3. *Given a rooted graph (G, r) with $\Delta(G) \leqslant 3$ and $\deg(r) \leqslant 2$, there is an optimal solution to the firefighter problem in which the vertex defended at each time has a burning neighbour.*

Proof. If $\deg(r) = 1$, then the lemma is trivial, so we consider the case where r has two neighbours x_1 and x_2. Suppose the statement is false and let (G, r) be a minimal counterexample.

Suppose there is an optimal strategy in which the first vertex defended is a neighbour of r, without loss of generality say x_1. Then $(G - \{x_1, r\}, x_2)$ is a smaller counterexample, a contradiction.

Let u be the closest vertex to r which is defended in an optimal strategy σ, and suppose $\operatorname{dist}(u, v) \geqslant 2$. If two neighbours of u are burning at the end of the strategy, then u lies on a cycle which is completely burned except for u. In this case, Strategy 1 saves at least as many vertices as σ. If no neighbour of u is burned, then $\operatorname{dist}(u, r) = \infty$, in which case σ is clearly not optimal. It remains to consider the case where exactly one neighbour of u is burned. By defending this neighbour instead of u, and leaving all other moves the as before, u will still not burn and the new strategy will save one more vertex. This proves the lemma. □

Theorem 4. *Strategy 1 yields an optimal solution to the firefighter problem on a rooted graph (G, r) with maximum degree at most three and such that $\deg(r) = 2$.*

Proof. By Lemma 3, there is an optimal solution σ in which each vertex defended has a burning neighbour. If the fire can no longer spread after it burns a vertex ℓ of degree one, then at least $\operatorname{dist}(l, v) + 1$ vertices are lost, in which case Strategy 1 is optimal. The same is true if the fire can no longer spread after only unburned neighbour u of a degree 2 vertex is defended, i.e. at least $\operatorname{dist}(u, v) + 1$ vertices are burned, and Strategy 1 does at least as well.

Suppose the fire can no longer spread because the second of two neighbours of a degree three burning vertex w is defended. Then, since each vertex defended has a burning neighbour, w is on a cycle which is completely burned except for the first of its two defended neighbours. In this case, Strategy 1 does at least as well (using Case 2).

Since $\Delta(G) \leqslant 3$ the above are the only three ways in which the fire can no longer spread. in each case, Strategy 1 is optimal. □

Corollary 5. *Let (G, r) be a rooted graph with maximum degree at most three and such that $\deg(r) = 2$. The maximum number of vertices that can be saved is $|V(G)| - \min\{f(x) \mid x \in (V_1 \cup V_2 \cup V_c)\}$.*

Corollary 6. *The firefighter problem is solvable in polynomial time for graphs with maximum degree three in which the fire starts at a vertex of degree at most 2.*

References

[1] A.S. Finbow, B.L. Hartnell, On designing a network to defend against random attacks on radius two, Networks 19 (1989) 771–792.
[2] S. Finbow, B. Hartnell, Q. Li, K. Schmeisser, On minimizing the effects of fire or a virus on a network, J. Combin. Math. Combin. Comput. 33 (2000) 311–322.
[3] P. Fogarty, Catching the fire on grids, M.Sc. Thesis, Department of Mathematics, University of Vermont, USA, 2003.

[4] M.R. Garey, D.S. Johnson, Computer and Intractability, A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, New York, 1979.
[5] G. Gunther, B. Hartnell, Graphs with isomorphic survivor graphs, Congr. Numer. 79 (1990) 69–77.
[6] A. Yousef, C. Gary, O.R. Oellermann, A.J. Schwenk (Eds.), Graph theory, combinatorics, and applications, vol. 2, Proceedings of the sixth quadrennial international conference on the theory and applications of graphs held at Western Michigan University, Kalamazoo, MI, USA, May 30–June 3, 1988, Wiley, New York, 1991, ISBN: 0-471-53219-3.
[7] B.L. Hartnell, Firefighter! An application of domination. Presentation, 24th Manitoba Conference on Combinatorial Mathematics and Computing, University of Manitoba in Winnipeg, Canada, 1995.
[8] B.L. Hartnell, W. Kocay, On minimal neighbourhood-connected graphs, Discrete Math. 92 (1991) 95–105.
[9] B. Hartnell, Q. Li, Firefighting on trees: how bad is the greedy algorithm? Proceedings of the Thirty-first Southeastern International Conference on Combinatorics, Graph Theory and Computing, Boca Raton, FL, 2000, Congr. Numer. 145 (2000) 187–192.
[10] G. MacGillivray, P. Wang, On the firefighter problem, J. Combin. Math. Combin. Comput. 47 (2003) 83–96.
[11] S.A. Moeller, P. Wang, Fire control on graphs, J. Combin. Math. Combin. Comput. 41 (2002) 19–34.