A New Survey on the Firefighter Problem

by

Connor Wagner
B.Sc., University of Victoria, 2017

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Mathematics and Statistics

© Connor Wagner, 2021
University of Victoria

All rights reserved. This thesis may not be reproduced in whole or in part, by photocopying or other means, without the permission of the author.

We acknowledge with respect the Lekwungen peoples on whose traditional territory the university stands, and the Songhees, Esquimalt, and W̱SÁNEĆ peoples whose historical relationships with the land continue to this day.

A New Survey on the Firefighter Problem

by

Connor Wagner
B.Sc., University of Victoria, 2017

Supervisory Committee

Dr. Gary MacGillivray, Co-Supervisor
(Department of Mathematics and Statistics)

Dr. Kieka Mynhardt, Co-Supervisor
(Department of Mathematics and Statistics)

ABSTRACT

Firefighter is a discrete-time dynamic process that models the spread of a virus or rumour through a network. The name “Firefighter” arises from the initial analogy being the spread of fire among the vertices of a graph. Given a graph G, the process begins at time $t = 0$ when one or more vertices of G spontaneously “catch fire”. At each subsequent time step, a collection of $b \geq 1$ “firefighters” defend a set of vertices which are not burning, and then the fire spreads from each burning vertex to all of its undefended neighbours.

There are many possible objectives one could have, for example minimizing the expected number of vertices burned when the fire breaks out at a random location or locations, finding the maximum number of vertices that can be saved from burning if the fire breaks out at known locations, minimizing the length of the process, or bounding the proportion of vertices that can be saved from burning. It is also possible to consider multiple objectives that may be in conflict. There are a great number of papers in the literature which address these, and other, issues in terms of computational complexity, algorithms, approximation, asymptotics, heuristics, and more.

The main purpose of this thesis is to survey developments on Firefighter and its variants which have appeared in the literature subsequent to a previous survey that appeared in 2009 [S. Finbow and G. MacGillivray. The firefighter problem: A survey of results, directions and questions. *Australas. J. Comb.*, 43, 2009]. The thesis concludes with a list of open problems and future directions from the previous survey, annotated with references for papers that have made progress on those topics since then.

Table of Contents

List of Figures

ACKNOWLEDGEMENTS

I would like to thank my supervisors, Gary MacGillivray and Kieka Mynhardt, for their unwavering support and encouragement, as well as for their understanding and patience. I would also like to thank my mother and father for all the ways they have supported my education thus far, and for always being willing to listen and offer timely advice. I would also like to thank my fellow graduate students and everyone in the department for giving me a sense of community, and in particular Aaron Slobodin, for helping me with my thesis template and my C.V. Lastly, I would like to thank my wife Amber, and my children, for being understanding while I dedicated long hours to the demands of my schooling, and for giving me purpose in my life.

DEDICATION

For Jane and Frank Wagner.

Chapter 1

Introduction

Firefighter is a discrete-time process that takes place on a simple graph. It models the spread of a fire from one or more vertices, with one or more firefighters at each time step permanently defending selected vertices from catching fire. If we view the vertices of a graph as individuals or groups of individuals (such as households, social groups, etc.), the "fire" as a virus or contagion, and the firefighters as health-care professionals administering a perfectly effective vaccine, then Firefighter can be used to model the spread of disease in a community. Other interpretations include the spread of gossip or information in a social or computer network, and the spread of invasive species like Asian carp in the Mississippi river system.

Given a graph G, imagine that at time 0 a fire breaks out at the vertices of a non-empty set $F \subseteq V(G)$. A vertex is called *burned* or *burning* if it is on fire at any time. At each time $i \geq 1$ a set of b firefighters chooses a subset of at most b non-burning vertices to *defend*, thus preventing them from burning. Defended vertices remain defended forever. The fire then spreads from all burning vertices to all of their non-burning, undefended neighbours. The process continues until the fire can no longer spread.

Firefighter was first studied in a paper by Finbow, Hartnell, Li, and Schmeisser [48]. The results were first presented at the 25th Manitoba Conference on Combinatorial Mathematics and Computing in 1995 [61]. Much of the literature points to that conference as the origin of what has come to be called *Firefighter*. For example see [10, 50, 95].

Firefighter is illustrated in Figure 1. In this example, $F = \{r\}$ and there is $b = 1$ firefighter. In this figure, at time zero, the fire breaks out at r, which is coloured red to indicate that it is burning, and at each subsequent time step a vertex is defended and coloured black. Then the fire spreads to the remaining undefended neighbours of burning vertices, which are coloured red. In this example, after $t = 2$ the fire can no longer spread.

There are a number of different possible objectives one can pursue with respect to Firefighter.

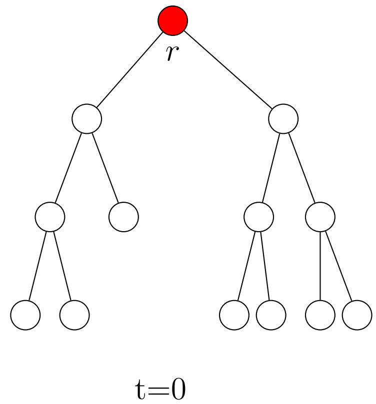

Figure 1.1: The fire breaks out at vertex r, black square vertices are defended, red vertices are burning.

Objectives commonly appearing in the literature include:

- Save the maximum possible proportion of vertices (finite or infinite graphs);
- Contain the fire (stop any further vertices from burning; terminate the process) in a finite number of time steps (infinite graphs);
- Determine whether it is possible to save a specific subset of the vertices of a graph with a given number of firefighters;
- Find the minimum number of firefighters needed to contain a fire or save a specific number, proportion or subset of the vertices;
- Design or characterize graphs where the expected number of vertices burned (averaged over all possible sets of initial fires F of fixed cardinality) is minimized;
- Assuming that for directed graphs the fire can only spread in the direction of the arcs, choose an orientation of the graph that minimizes the maximum number of vertices burned under an optimal strategy when the set F of vertices initially on fire is chosen after the orientation has been chosen;
- Find solution strategies for the "Multiobjective firefighter problem", in which there are multiple (possibly conflicting) objectives.

A survey of results on Firefighter and its variants appeared in 2009 [50]. The purpose of this thesis is to provide an update of that survey. There is some overlap with its content for consistency, context and completeness.

Relevant definitions that are used throughout the thesis are introduced in Chapter 2. Definitions that are used only locally are introduced when necessary. Subsequently, a framework is introduced which unifies a large number of variations of Firefighter, specifically those in which the way the process is carried out is not changed. The chapter concludes with a discussion of decision problems, easy examples to help the reader check their understanding, and a discussion of conflicting objectives.

Results related to computational complexity are the main focus of Chapter 3. The first section reports on NP-completeness that generalized earlier ones to more general versions of Firefighter, and the next does the same for greedy algorithms. Integer linear programming has been used to compute solutions to some versions of Firefighter, both with respect to saving the maximum possible number of vertices, or ending the process in the smallest number of time steps. Results of this type are discussed in the next sections, as is the *integrality gap* – the ratio between the optimum solution to an integer linear program and its linear programming relaxation. Attention then shifts to fixed parameter tractability (FPT), the various FPT algorithms that have been used, and the graph classes on which they are effective. The last section of the chapter is a discussion of some heuristic approaches that have been proposed and investigated.

Arguably the most studied aspect of Firefighter is the so-called surviving rate which, roughly speaking, is the expected proportion of vertices that can be saved from a random outbreak of fire. The literature contains a substantial number of results, most of which are concerned with restricted families of planar graphs or graphs with bounded average degree. These are surveyed in Chapter 4.

The focus of Chapter 5 is variants of Firefighter that have not been discussed elsewhere in the thesis. These include versions of Firefighter where the way the process is carried out is changed in some way, or the process takes place on a directed graph, or the process is continuous-time rather than discrete-time.

In the last section of the thesis we list the 26 open problems and suggestions for future research that were given at the conclusion of the first survey and, for each, give references to the relevant papers which have appeared.

Chapter 2

Definitions and Versions

Some of the definitions and terminology needed to discuss Firefighter are introduced in this section. We also introduce a unified framework which captures most variations of Firefighter that have arisen in the literature.

2.1 Definitions

For a positive integer f, an f*-rooted graph* is an ordered pair (G, F), where G is a graph and $F \subseteq V(G)$ is an f-element subset (or f-subset) of vertices called *roots*. A *rooted graph* is a graph which is f-rooted for some f. When $F = \{r\}$ we write (G, r) rather than $(G, \{r\})$.

To more formally introduce Firefighter, consider the following discrete-time colouring process. Let f and b be positive integers. Let (G, F) be an f-rooted graph whose vertices are initially uncoloured. At time $t = 0$, colour all the vertices in F red. At each time $t \geq 1$, colour b or fewer uncoloured vertices black, then colour red each uncoloured vertex with a red neighbour. The process terminates at the first time step i when the set of red vertices after step i is equal to the set of red vertices after step $i-1$. A vertex that is coloured red is called *burning* or *burned*. It is easy to observe that the set of red vertices at the end of each step induces a subgraph with at most $f = |F|$ components, and if G is connected and there are any uncoloured vertices at the end of the process, the set of black vertices is a vertex cut. A vertex is called *protected* or *defended* if it is coloured black, and *saved* if it never burns (i.e. it is not red when the process ends).

Given a graph G, a set F of initial fires, b firefighters, and $i \geq 0$ completed rounds of the firefighter process, we define a *valid move* to be a placement of firefighters on b or fewer nonburning, unprotected vertices at a time step i. We define a *strategy* to be a sequence of valid moves for each time $i \geq 1$.

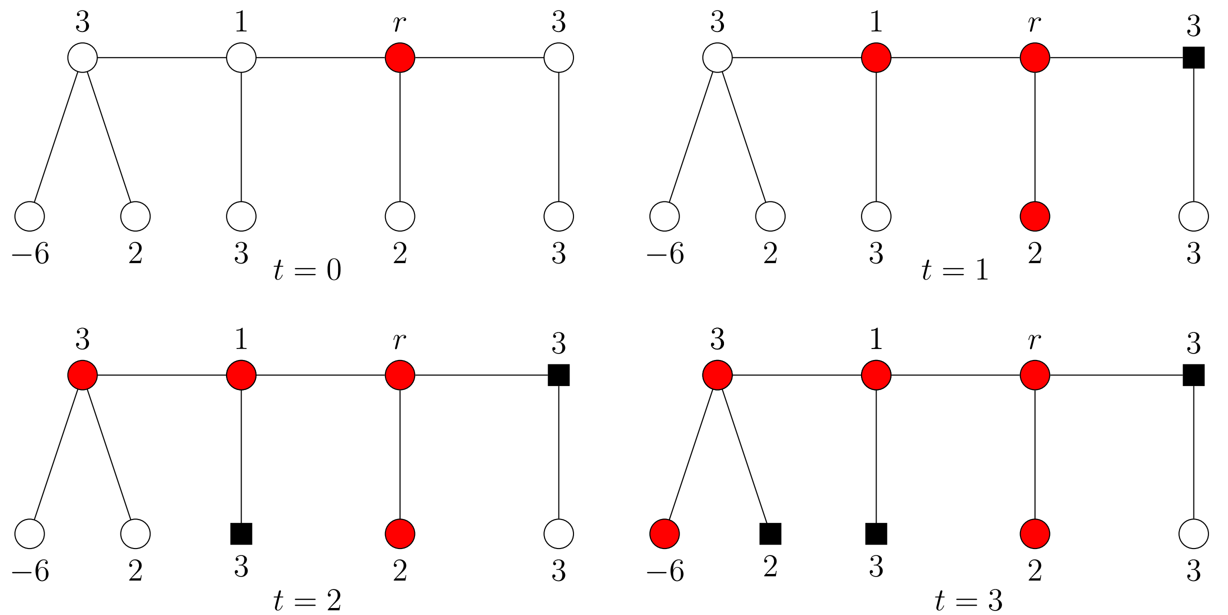

Figure 2.1: An example of Firefighter on a weighted caterpillar. Here the objective is to maximize the total weight of the saved vertices. An optimal strategy allows the negative-weighted vertex to burn.

2.2 A Unified Framework

We now introduce the unified framework mentioned above.

A *weighted* f-rooted graph is a triple (G, F, w), where (G, F) is an f-rooted graph and $w : V(G) \to \mathbb{R}$ is a weight function. If $V(G) = \{v_1, v_2, ..., v_n\}$, we also write w_i for $w(v_i)$. Negative weights are permitted. They correspond to an incentive to let vertices burn rather than to save them. Negative weights appear in [30], are briefly discussed in Section 3.1 (see Theorem 3.1.5), appear in Figure 2.1, and otherwise will not arise in this thesis.

Given a finite weighted f-rooted graph (G, F), a weight function $w : V(G) \to \mathbb{R}$, and b firefighters available at each time step, we define *Max-Weighted-b-Fire* to be the problem of saving a subset of vertices of maximum weight by defending b or fewer vertices at each time step as in Firefighter.

We now explain how, by carefully selecting the weight function w, Max-Weighted-b-Fire encompasses several variants of Firefighter.

In the original version of Firefighter, a single vertex r catches on fire at the beginning of the process, that is, $F = \{r\}$. There is one firefighter available at each time step. The goal is to maximize the number of vertices that never burn. This corresponds to the version of Max-Weighted-b-Fire in which $b = 1$ and each vertex of G has weight 1.

There is also an optimization problem called Max (S, b)-Fire (see [9]). Here, there is one initial fire, though it easy to generalize to an outbreak of size $f \geq 2$. The goal is to save as many vertices as possible from a specific set S with b firefighters per round. This problem corresponds to the restriction of Max-Weighted-b-Fire in which $f = 1$, b is some fixed integer, the vertices in S are assigned weight 1, the vertices not in S are assigned weight 0, and the goal is to maximize the total weight of the vertices that can be saved. In this version, the number of vertices from $V(G) \setminus S$ burned or saved at the end of the process is irrelevant.

The following proposition is self-evident:

Proposition 2.2.1. *If H is a spanning subgraph of G, then, if an outbreak of fire occurs at the same vertex of both graphs, then one can save at least as many vertices in H as they can save in G.*

2.3 Decision Versions

In this section, we formalize decision versions of the problems discussed in the previous section. We begin by describing the decision version of Max-Weighted-b-Fire.

For positive integers f and b, the decision problem WEIGHTED-b-FIRE is defined as follows.

INSTANCE: A weighted f-rooted graph (G, F, w) and a real number B.

QUESTION: If at most b vertices are defended at each time step, is it possible to save a subset $X \subseteq V(G)$ of vertices such that $\sum_{x \in X} w(x) \geq B$?

By setting analogous restrictions, as before, for WEIGHTED-b-FIRE, we have the corresponding decision versions of Firefighter and Max (S, b)-Fire, the latter of which includes a well-studied specific instance called S-Fire.

The decision version of Firefighter is the version of WEIGHTED-b-FIRE where $f = b = 1$, and each vertex has weight 1.

FIREFIGHTER

INSTANCE: A weighted f-rooted graph (G, F, w) such that $w(x) = 1$ for all $x \in V(G)$, and a real number B.

QUESTION: If at most one vertex is defended at each time step, is it possible to save a subset $X \subseteq V(G)$ of vertices such that $\sum_{x \in X} w(x) \geq B$?

FIREFIGHTER admits a natural extension to b-FIREFIGHTER, in which b firefighters are available at each time step.

The decision problems (S, b)-FIRE and S-FIRE are formalized below.

(S, b)-FIRE

INSTANCE: A rooted graph $(G, \{r\})$ and a subset $S \subseteq V(G)$.

QUESTION: Can b firefighters save all vertices in S?

The correspondence with WEIGHTED-b-FIRE is as discussed above. Restrict to $f = 1$ and $w(v) = 1$ for $v \in S$ and $w(v) = 0$ for $v \in V(G) \setminus S$

S-FIRE

INSTANCE: A rooted graph (G, r) and a subset $S \subseteq V(G)$.

QUESTION: Can one firefighter save all vertices in S?

The correspondence with WEIGHTED-b-FIRE is as above, except now $b = 1$.

2.4 Examples Involving Saving Vertices

One possible objective in Firefighter is to save the maximum number of vertices. Let $MVS(G, r)$ denote the maximum number of vertices of the finite graph G that can be saved when there is an outbreak at r and a single firefighter available. We now give examples of five classes of graphs G for which it is easy to determine $MVS(G, r)$.

1. *For* $n \geq 3$,

$$MVS(P_n, r) = \begin{cases} n-1 & \text{if r is an endpoint of the path} \\ n-2 & \text{otherwise.} \end{cases}$$

2. *For* $n \geq 3$, $MVS(C_n, r) = n - 2$.
3. *For* $n \geq 2$, $MVS(K_n, r) = 1$.
4. *For* $n \geq 2$, $MVS(K_{n,n}, r) = 2$.
5. *For* $n \geq 2$, $MVS(Q_n, r) = n$, where Q_n denotes the n-cube [78].

It is interesting to note that, as seen in (5) above, it is not always possible to save a large proportion of the vertices even if the average degree is relatively low when compared to the number of vertices. (The n-cube has 2^n vertices and is n-regular.)

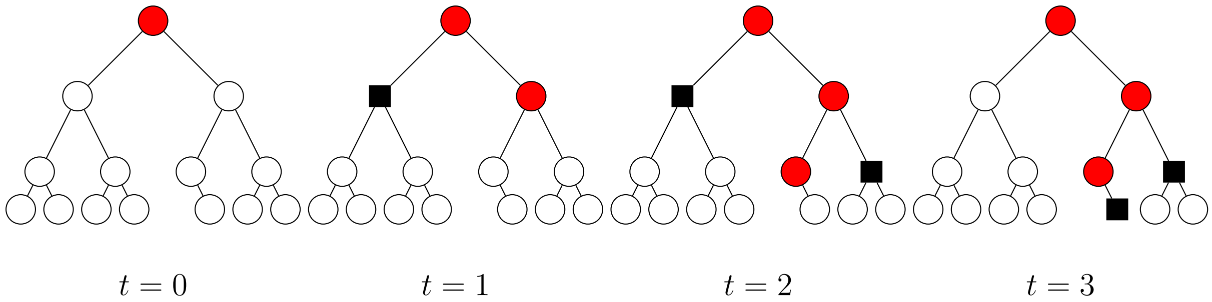

Figure 2.2: Strategy illustrated in Proposition 2.5.1

2.5 Equivalent and Conflicting Objectives

What constitutes a strategy for defending a graph is defined in Section 2.1. In this section, we give a more elaborate discussion of strategies.

We define an *optimal strategy* with respect to a given objective to be a strategy which gives best possible results with respect to the given objective.

For a class of graphs $\mathcal{G}$, we define two objectives, A and B, to be *equivalent under* $\mathcal{G}$ if for any $G \in \mathcal{G}$ there is a strategy that is optimal for both A and B. If two objectives are equivalent under the class of all graphs, we simply call them equivalent. If two objectives are not equivalent (under $\mathcal{G}$) then they are *in conflict* (under $\mathcal{G}$).

We now show that it is possible for two objectives that are in conflict in the class of all graphs to be equivalent in a more restricted class of graphs.

Proposition 2.5.1. *Let A and B be the objective of minimizing the number of vertices burned, and the objective of terminating the spreading process in the minimum number of time steps, respectively. For binary trees where the fire breaks out at the root, the following strategy produces an optimal strategy for A and B.*

1. *Defend only vertices with burning neighbours.*
2. *At each time, defend a vertex u subject to the above restriction for which the distance to a descendant of degree at most 2 is maximized.*

Figure 2.2 illustrates the implementation of the strategy on a binary tree.

Figure 2.3 illustrates a tree where the two objectives A and B, above, are in conflict. In this tree, with an initial fire at r, there is no strategy that simultaneously saves the most vertices and terminates the process in the fewest number of time steps.

The focus of the research on Firefighter and its variants depends largely on the class of graphs being studied. For instance, the list of possible objectives for Firefighter on finite graphs differs from the possible objectives for infinite graphs (see Chapters 1 and 4).

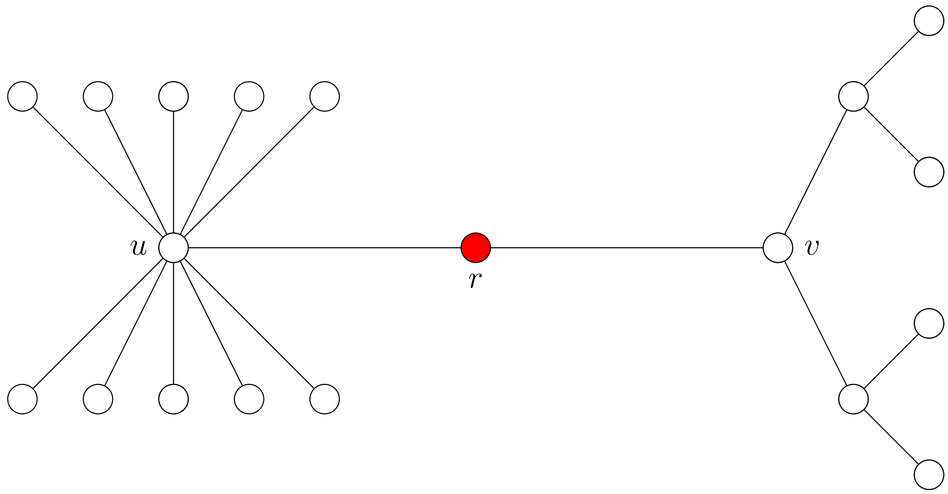

Figure 2.3: The fire breaks out at vertex r, defending u first saves the most vertices, while defending v first contains the fire in fewer time steps.

Equivalent or conflicting objectives are important when one considers the "multi-objective firefighter problem" as discussed in Subsection 3.5.2. Decisions must be made about how to prioritize conflicting objectives.

Chapter 3

Algorithms and Complexity

Not long after the introduction of Firefighter, the study of its computational complexity began. Many versions of Weighted-b-Fire are computationally "hard", while some restrictions of those are computationally "easy". These results led researchers to consider FPT algorithms, approximation algorithms, and heuristics.

3.1 NP-completeness Results

The complexity of FIREFIGHTER was first studied in 2003 [78]. The following later results were included in Finbow and MacGillivray's 2009 survey [50].

Theorem 3.1.1. [49] FIREFIGHTER *is NP-complete for rooted trees with maximum degree* 3 *in which the root has degree* 3*.* FIREFIGHTER *is solvable in polynomial time for graphs of maximum degree* 3 *if the root has degree at most* 2*.*

See Figure 3.1 for an example of the polynomial-time algorithm mentioned above.

In addition, Cai and Wang [20] present an algorithm that solves Firefighter on rooted trees to optimality and that runs in $2^{O(\sqrt{n}\log n)}$ time.

Further, FIREFIGHTER is NP-complete for cubic graphs [66]. By contrast, Costa, Dantas, and Rautenbach [32] proved that FIREFIGHTER is solvable in polynomial time for Blanusa, Flower, and Goldberg snarks. (Snarks are simple, connected, bridgeless, cubic graphs with chromatic index equal to 4.)

Theorem 3.1.1 implies that FIREFIGHTER is NP-complete when restricted to rooted planar or outerplanar graphs with maximum degree 3 in which the root has degree 3. It also implies that the corresponding directed version is NP-complete for acyclic orientations of graphs with maximum degree 3 in which the root has out-degree 3.

The following result by King and MacGillvray [66] was also mentioned in the survey [50].

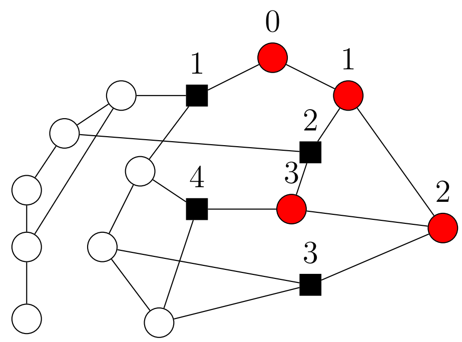

Figure 3.1: Given a graph with maximum degree 3 and a root of degree 2, a polynomial-time strategy for finding an optimum solution for Firefighter dictates steering the fire to a vertex of degree at most 2, or, if not possible, around the shortest cycle possible. The numbers correspond to the time step in which each vertex is defended or burned.

Theorem 3.1.2. [66] S-FIRE *is NP-complete, even if* S *is the set of leaves of a rooted tree of maximum degree* 3 *in which the root has degree* 3*.* S-FIRE *is solvable in polynomial time for rooted trees of maximum degree* 3 *if the root has degree at most* 2*.*

In 2013, Bazgan, Chopin and Ries generalized Theorem 3.1.2 as stated below.

Theorem 3.1.3. [9] *Let* $b \geq 1$*,*

1. (S, b)-FIRE *is NP-complete for trees of maximum degree* $b + 2$ *even when* S *is the set of leaves.*
2. MAX-(S, b)-FIRE *is NP-hard for trees of maximum degree* $b+3$ *even when* S *is the set of all vertices.*
3. (S, b)-FIRE *and* MAX-(S, b)-FIRE *are solvable in polynomial time for rooted trees of maximum degree* $b + 2$ *when the fire breaks out at a vertex of degree at most* $b + 1$*.*
4. MAX-(S, b)-FIRE *is solvable in polynomial time on rooted* k*-extended-caterpillars.*

In this paper, the authors define a k-caterpillar as a tree in which every vertex is within distance k of a central path. We will refer to these here as k*-extended-caterpillars*.

Corollary 3.1.3.1. *Let* $b \geq 1$ *be an integer, and let* $f = 1$*. Then* WEIGHTED-b-FIRE *is NP-complete for rooted trees of maximum degree* $b+2$*, even for the restriction where the root has degree* $b + 2$ *and all weights are positive.*

In contrast to Theorem 3.1.1, Duffy proved that:

Theorem 3.1.4. [42, 43] S-FIRE *is NP-complete even when restricted to graphs of maximum degree* 3 *where the fire breaks out at a vertex of degree* 2*.*

Finbow and MacGillivray [50] speculated that WEIGHTED-1-FIRE may be solvable in polynomial time for the class of graphs of maximum degree 3 if the fire breaks out at a vertex of degree 2. This is false unless $P = NP$.

Theorem 3.1.5. [9, 41] Weighted-1-Fire *is NP-complete when restricted to complete binary trees of height* 3*, if negative weights are allowed. The problem is solvable in polynomial time when restricted to binary trees in which all weights are non-negative.*

3.2 Greedy Algorithms and Firefighting on Trees

A *greedy algorithm* is a decision-making algorithm that makes an optimal choice with regard to some (usually local) criteria at each step, regardless of whether or not the sequence of choices necessarily results in an optimal solution. Greedy algorithms are often much simpler and require fewer computational resources than algorithms that find globally optimal solutions. Greedy algorithms can produce excellent or even optimal solutions for different types of graph theory problems. For example, Kruskal's algorithm [73] and Prim's algorithm [93] find a minimum spanning tree of any connected, undirected edge-weighted graph. In this section we discuss some greedy algorithms for Firefighter on trees.

Consider a graph where every vertex has weight 1, and apply the greedy algorithm where the firefighter chooses to defend the undefended, unprotected vertex of highest degree. This seems like an acceptable short-term solution as it results in the most vertices being saved in the first time step. However, it is easy to construct a tree where the number of vertices saved by this greedy strategy is arbitrarily far away from the number saved in an optimum strategy in both relative and absolute terms. See Figure 3.2.

There is a greedy algorithm for Firefighter that always saves at least half of the maximum number of vertices of a tree [62]. If we restrict the class of graphs even further, to caterpillars, then there is a greedy strategy that saves the maximum number of vertices [78].

Let $f = b = 1$ and consider a tree with all vertices of weight 1. The following algorithm, introduced by Hartnell and Li, will always save at least half of the maximum possible number of vertices [62]. (An algorithm that saves a proportion of at least ϵ of the optimal number of vertices is called an *ϵ-approximation algorithm.*) Note, however, that the number or proportion of vertices burned by the algorithm, when compared to an optimal strategy, can be arbitrarily large.

Strategy 3.2.1. *Given a rooted tree T with root r which is the original source of the fire, consider all of the immediate neighbours of r, call them n_1, n_2, ...*

For $i = 1, 2, ...,$ let N_i be the number of descendants of n_i.

1. *Defend the vertex n_i with the largest corresponding number N_i. (Break ties arbitrarily).*
2. *For subsequent time steps, let m_1, m_2 ... be neighbours of burning vertices. Then each m_i is the unique vertex lying on every path between burned vertices and a set of*

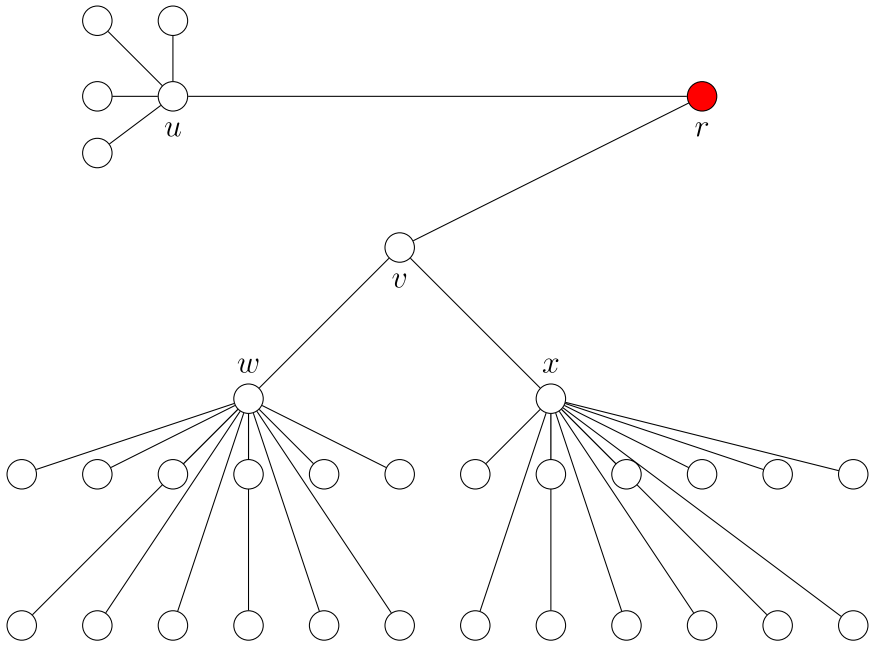

Figure 3.2: Fire breaks out at r. The degree-greedy algorithm dictates saving u first, but only saving v first can prevent both w and x from catching fire. By increasing the number of neighbours of both x and w, one can find a tree where the degree-greedy algorithm performs arbitrarily poorly compared to an optimal strategy in relative or absolute terms

descendants of cardinality M_i. Defend the vertex m_i with the largest corresponding M_i of largest cardinality.

3. *Stop when the set of burned vertices is the same in two consecutive time steps.*

See Figure 3.3 for an example of how the greedy algorithm based on Strategy 3.2.1 works, and a comparison to an optimal strategy on the same tree.

It is easy to observe that if a single fire breaks out in a rooted tree, then any strategy that saves the maximum possible number of vertices only defends vertices with burning neighbours. Surprisingly, this is not always the case in f-rooted trees when $f \geq 3$. The following results are due to Costa, Dantas, Dourado, Penso, and Rautenbach [29].

Theorem 3.2.1. [29] *Let $b \geq 1$ be an integer. If T is an f-rooted tree with $f \leq 2$, then the maximum possible number of vertices can be saved by a strategy that only defends vertices with burning neighbours.*

Theorem 3.2.2. [29] *There exist integers $f \geq 3$ and $b \geq 1$, and an f-rooted tree (T, F) such that b firefighters cannot save the maximum possible number of vertices using a strategy that saves only neighbours of burning vertices.*

See Figure 3.4 for an example of a 3-rooted tree where an optimal strategy requires defending a vertex not adjacent to a vertex which is on fire. Also, for cyclic graphs with $f = b = 1$,

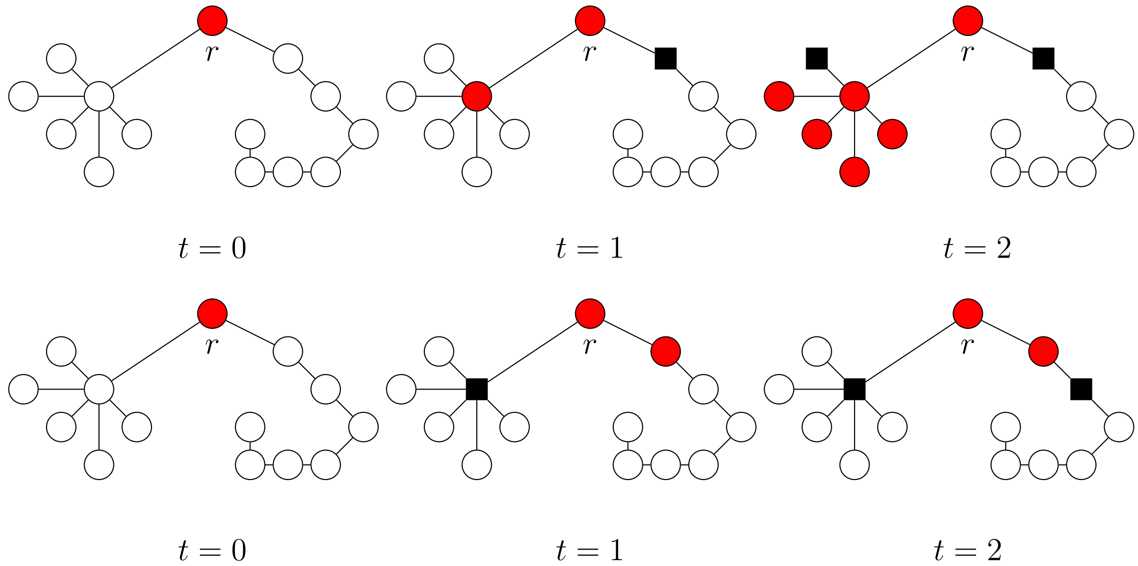

Figure 3.3: The top row illustrates the sequence of moves dictated by the greedy strategy of saving the vertex whose defence saves the largest subtree. The bottom row illustrates an optimal solution. The fire breaks out at r.

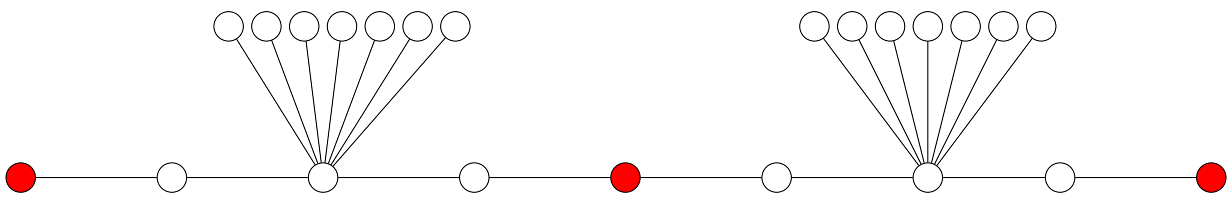

Figure 3.4: Let $b = 1$. In order to save the 3rd and 7th vertices from the left one must first defend one of these vertices in round 1, and then defend the other in round 2. Any other defence sequence results in one of these vertices being burnt and all of its undefended neighbours being burnt in the following round.

an optimal strategy can also require defending vertices not adjacent to a vertex which is on fire. See Figure 3.5.

3.3 Integer Linear Programs, Containment, and Approximations

In 2002, Wang and Moeller described a strategy in which two firefighters can contain an outbreak of a single fire on the infinite Euclidean plane in eight time steps with 18 vertices burned [87]. They proved that this was the minimum number of time steps needed, but did not, however, make a claim as to whether or not this was the minimum number of burnt vertices possible.

Fogarty [52] provided a proof of a condition for a necessary minimum number of firefight-

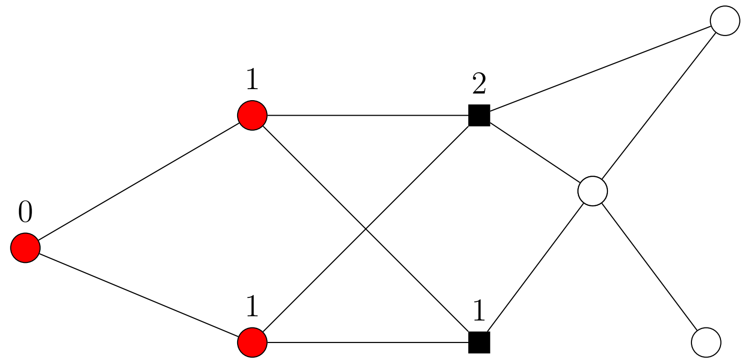

Figure 3.5: An example of how an optimum solution for the firefighter problem on a cyclic graph with $f = b = 1$ may need to involve defending vertices not immediately threatened by the fire. The numbers correspond to the time step in which the vertex is defended or burned.

ers required for containment of a graph. Loosely speaking, the theorem states that if the number of vertices of increasing distance from the outbreak grows faster than the number of firefighters available at each time step, then the fire cannot be contained. Dyer, Martinez-Pedroza and Thorne [44] proved that if the number of vertices at increasing distance from the outbreak grows at a rate slower than $O(n^2)$ per unit increase in distance (this is called quadratic growth), then with enough firefighters, a single outbreak, or any finite outbreak can be contained.

Costa, Dantas, and Rautenbach [31] proved that for f-rooted trees and one firefighter, the proportion of vertices that can be saved is greater than $(\frac{1}{3})^f$ and that for an outbreak of two fires the proportion of vertices saved is greater than $\frac{4}{9} - o(1)$.

Chen, Hu, Wang and Zhang [24] studied fire containment on infinite k-dimensional square grids with a variant of Firefighter. In this version, named "continuous firefighting", the firefighter may only move from a vertex to its immediate neighbour in each round of fire-fighting. The authors proved that for a single fire, and a square grid of dimension k, the minimum number of firefighters needed to contain an outbreak of a single fire in the continuous firefighting model was $2k$, (this number of firefighters is both necessary and sufficient for containment) and that in the 2-dimensional grid, 5 firefighters were necessary and sufficient to contain any outbreak of multiple fires regardless of the number of initial fires or distance between points of outbreak.

Develin and Hartke [38] developed an integer linear program (ILP) for $MVS(G, F)$ that verified that Wang and Moeller's solution [87] was optimal in terms of minimizing the number of vertices burnt. Finbow and MacGillivray [50] presented an integer linear program which is only slightly modified from Develin and Hartke's 2007 ILP:

$$\text{Min} \sum_{x \in V} b_{x,T}$$

$$\text{Subject to: } \begin{cases} b_{x,t} \geq b_{x,t-1} & x \in V, 1 \leq t \leq T \\ d_{x,t} \geq d_{x,t-1} & x \in V, 1 \leq t \leq T \\ b_{x,t} + d_{x,t} \geq b_{y,t-1} & x \in V, y \in N(x), 1 \leq t \leq T \\ b_{x,t} + d_{x,t} \leq 1 & x \in V, 1 \leq t \leq T \\ \sum_{y \in N(x)} b_{y,t-1} \geq b_{x,t} & x \in V, 1 \leq t \leq T \\ \sum_{x \in V} (d_{x,t} - d_{x,t-1}) \leq d & 1 \leq t \leq T \\ d_{x,0} = 0 & x \in V \\ b_{x,0} = \begin{cases} 1 & \text{if } x = r_i \text{ for some } i, 1 \leq i \leq f \\ 0 & \text{otherwise} \end{cases} & x \in V \\ b_{x,t}, d_{x,t} \in \{0,1\} & 1 \leq t \leq T \end{cases}$$

The Linear Programming (LP) relaxation of an ILP is a linear program in which the variables can take values that are not integers. Since there are more possible values (including the original integer values) for the variables, the optimum is always at least as great as the integer optimum. Often there is a difference between the optimum solution of an ILP and its LP relaxation. This difference is measured by a quantity called the integrality gap. Formally, the integrality gap is the ratio between the ILP optimal solution and the optimum solution to its LP relaxation. Note that this number is always less than or equal to 1 for Firefighter. Non-integer variables correspond to providing a fraction of protection to a vertex, often resulting in solutions that are not possible in Firefighter. This practice of defending portions of vertices is referred to as *Fractional Firefighting* [52]. Since we are often concerned with the optimal solution to Firefighter, it is of particular interest to study the integrality gap between the ILP optimum and the LP optimum.

In 2008, Cai, Verbin and Yang [20] used an LP relaxation of MacGillivray and Wang's 0-1 ILP [78] to improve upon Hartnell and Li's $\frac{1}{2}$-approximation algorithm described in Strategy

3.2.1:

$$\text{Max} \sum_{v \in V} w_v x_v$$

$$\text{Subject to:} \begin{cases} x_r = 0 & \\ \displaystyle\sum_{level(v)=i} x_v \leq 1 & \text{for each level } i \\ x_v + \displaystyle\sum_{\text{all ancestors of } v} x_u \leq 1 & \text{for every leaf } v \text{ of } T \\ 0 \leq x_v \leq 1 & \text{for every vertex } v \text{ of } T \end{cases}$$

Theorem 3.3.1. [20] *There is a polynomial-time* $(1 - 1/e)$*-approximation algorithm for Firefighter on trees.*

Note that $(1 - 1/e) \approx 0.632$. Iwaikawa, Kamiyama, and Matsui [64] further improved the approximation constant to 0.7144 for ternary trees.

Hartke [60] added additional constraints to MacGillivray and Wang's ILP in an attempt to narrow the integrality gap between the ILP optimal solution and the linear programming optimal solution. A computational study on trees showed that the LP relaxation of the augmented ILP found the optimum integer solution more often.

Costa, Dantas, Dourado, Penso, and Rautenbach [29] extended the $(1-1/e)$ -approximation algorithm to trees with arbitrary many firefighters and initial fires:

Theorem 3.3.2. [29] *Let* f *and* b *be two positive integers, let* $|F| = f$*, and let* T *be a tree. There is a polynomial-time* $(1-1/e)$*-approximation algorithm that approximates* $MVS(T, F)$*.*

In 2017, Chalermsook and Vaz [23] presented an example of a specially constructed graph in which the integrality gap was $\frac{5}{6}$ for Hartke's 2007 linear program.

3.4 Fixed Parameter Tractability

Parameterized complexity is a tool for approaching problems which currently require algorithms with superpolynomial running time. One can categorize problems as *Fixed Parameter Tractable* (FPT) if the solution can be computed in a time that is polynomial in the input size and superpolynomial in another parameter k. If k is fixed (and therefore bounded) then, for small values of k, one may be able to compute solutions with a reasonable amount of computational resources. With respect to Firefighter, the parameter k can be a number of different things, such as the number of leaves in a tree, the number of vertices burned, the number of saved vertices, or the number of defended vertices [20, 95].

We now introduce some terminology related to parameterized complexity.

A *parameterized problem* is a decision problem in which some parameter k associated with the problem is fixed. An *instance* of a parameterized problem consists of a pair (I, k), where I is the input, and k is the parameter. A parameterized problem is *fixed-parameter-tractable (FPT)* if it admits an algorithm that runs in $f(k)|I|^{O(1)}$ time for some computable[1] function f which does not depend on the input I. A *kernelization* for a parameterized problem is a polynomial-time algorithm that transforms an instance (I, k) into an instance (I', k') such that

(i) $|I'| \leq g(k)$ for some computable function g,

(ii) $k' \leq k$, and

(iii) (I, k) is a "Yes"-instance if and only if (I', k') is a "Yes"-instance. The pair (I', k') is called a *kernel* of (I, k), and a *polynomial kernel* if $g(k) = k^{O(1)}$.

The existence of a kernel (polynomial or otherwise) implies the existence of an FPT algorithm, and *vice versa*. A kernel replaces the input I with another input I' of size that is bounded above by a polynomial $g(k)$ involving only k and not n. Hence, since k is fixed, one can solve the instance (I', k') efficiently, and thus obtain the solution to the original instance (I, k).

Scott, Stege, and Zeh [95] were the first authors to use an FPT algorithm for a variant of Firefighter. In this version, called *Politician's Firefighting*, which commonly has one initial fire, the number of firefighters is not fixed for each round, but rather, in each round, there are equally as many firefighters as there are new burning vertices, with the following restriction: if a vertex v "generates" a firefighter by being on fire, this firefighter can only defend an unoccupied, non-burning neighbour of v. The goal of Politician's Firefighting is to save as many vertices as possible, as in Firefighter.

The decision version is defined as follows:

Politician's Firefighting

Instance: An f-rooted graph (G, F), a weight function $w : V(G) \to \mathbb{R}$ such that $w(x) = 1$ for all $x \in V(G)$, and a real number B.

Question: If, at each time step i, the number of vertices that can be defended is equal to the size of the set B_i of vertices burning at time i but not time $i-1$, each firefighter defends a neighbour of a vertex in B_i, and there is a one-to-one correspondence between the set of firefighters and the set of their burning neighbours in B_i, then is it possible to save a subset $X \subseteq V(G)$ of vertices such that $\sum_{x \in X} w(x) \geq B$?

[1] A function is computable if there exists an algorithm that, given an input of the function domain, returns the corresponding function value.

Because of the increased number of firefighters available, there turns out to be a polynomial time algorithm for politician's firefighting on trees [95]. The decision problem, Politician's Firefighting, is NP-hard even when restricted to planar graphs of maximum degree 5. There is an FPT algorithm for Politician's Firefighting on general graphs which runs in $O(m + k^{2.5}4^k)$-time, where m is the number of edges in the graph, and k is the number of vertices that burn.

Cai, Verbin, and Yang [20] found an FPT algorithm for Firefighter.

Theorem 3.4.1. [20]

(a) If k is the number of saved vertices, then the problem of deciding whether or not k vertices of a tree can be saved has an $O(k^2)$ kernel, and hence is FPT. The problem can be solved in $O(n) + 4^k k^{O(\log k)}$ time.

(b) If k is the number of saved leaves of a tree, then the problem of deciding whether or not k leaves of a tree can be saved has an $O(k^2)$ kernel, and hence is FPT. The problem can be solved in $O(n) + 2^{O(k)}$ time.

(c) If k is the number of defended vertices, then the problem of deciding whether or not a given number l of the vertices of a tree can be saved can be solved in $k^{O(k)}n$ time, therefore, the problem is FPT.

Note that in (c) the authors did not find a polynomial kernel.

While Cai, Verbin, and Yang [20] and Yang [104] found FPT results for Firefighter parameterized by the number of vertices saved, Leung [75] found that Firefighter was fixed-parameter-tractable (FPT) on general graphs when parameterized by the number of vertices burnt. The authors used a random separation technique from Cai, Chan, and Chan [18] to establish FPT algorithms on general graphs when parameterized by the number of burnt vertices, and degree-bounded and unicyclic graphs when parameterized by the number of vertices protected (the results in [20] only applied to trees).

Fixed parameter tractable problems all belong to a collection called W[1]. (See [40] for a formal definition of W[1].) It is known that FPT $\neq$ W[1] unless P=NP [40]. It is thus conjectured that FPT $\neq$ W[1].

Theorem 3.4.2. [8, 53, 54]

(a) [8] *If k is the number of saved vertices, then the problem of deciding whether or not k vertices can be saved is W[1]-hard, even when restricted to bipartite graphs.*

(b) [8] *If k is the number of vertices that are not saved, then the problem of deciding whether or not all but k vertices can be saved is W[1]-hard, even when restricted to bipartite graphs.*

(c) [8] *If k is the number of saved vertices, then the problem of deciding whether or not k vertices can be saved is FPT when restricted to planar graphs.*

(d) [53, 54] *If k is the number of saved vertices, then the problem of deciding whether or not k vertices can be saved is W[1]-hard when parameterized by the number of saved vertices.*

Chlebikova and Chopin [25, 26], noted that the proof of Theorem 3.1.1 used a tree with unbounded *pathwidth* [49], and refined the results as indicated below. A *path-decomposition* is a sequence of subsets of vertices of G such that the endpoints of each edge appear in one of the subsets and such that each vertex appears in a contiguous subsequence of the subsets, and the *pathwidth* is the minimum over all path decompositions of one less than the size of a largest subset in the sequence.

Theorem 3.4.3. [26]

1. Firefighter *is NP-complete on rooted trees of pathwidth 3.*
2. Firefighter *is FPT if parameterized by pathwidth.*
3. Firefighter *is FPT if parameterized by maximum degree Δ.*

The authors proved another dichotomous complexity result (the first being items 1 and 2 from the above theorem), namely that Firefighter is NP-complete on co-bipartite graphs (graphs whose complements are bipartite), but is FPT with respect to the parameter "cluster vertex deletion" which is the minimum number of vertices that have to be deleted to obtain a disjoint union of complete graphs. This quantity is easy to find for co-bipartite graphs.

Choudhari, Dasgupta, Misra, and Ramanujan [27] investigated the parameterized complexity of S-Fire. They proved that S-Fire is para-NP-hard (not fixed-parameter-tractable unless P=NP) when parameterized by the size of S. On the other hand, S-Fire is FPT on general graphs when parameterized by the number of firefighters b. By contrast, we saw in Theorem 3.1.4 that S-Fire is NP-complete when restricted to graphs of maximum degree 3 where the fire breaks out at a vertex of degree 2.

Das, Enduri, Kiyomi, Misra, Otachi, Reddy, and Yoshimura [36] investigated the fixed parameter tractability of Firefighter when it is parameterized by "distance" to specific classes of graphs such as threshold graphs, bounded diameter graphs, disjoint union of stars, and split graphs. The notion of *distance* here is defined to be the size of the set (called a *modulator*) of vertices that must be deleted for the resulting graph to belong to the graph class in question. They prove that Firefighter is W[1]-hard when parameterized by the size of a modulator to diameter at most 2 graphs and split graphs, but that it is FPT when parameterized by the size of a modulator to threshold graphs and disjoint unions of stars. They also found that Firefighter admits a polynomial kernel when parameterized by the size of a modulator to a clique but it does not admit a polynomial kernel when parameterized by the size of a modulator to a disjoint union of stars unless NP $\subseteq$ coNP/ poly.

3.5 Heuristics

While fixed-parameter-tractability gives options to researchers seeking to find alternative methods of finding optimal solutions of FIREFIGHTER, in recent years there has been a shift in the focus of how to approach computationally difficult instances of FIREFIGHTER. Many recent papers have investigated the use of heuristics in order to find approximations to optimal solutions in reasonable running times.

A *heuristic* is a technique for finding good solutions to computationally hard problems, usually by sacrificing optimality to a small degree. A *metaheuristic* is a procedure (which is usually itself a heuristic) that is designed to generate heuristics that find sufficiently accurate solutions to optimization problems. A *simheuristic* is an extension of metaheuristics that incorporates simulations into metaheuristic models.

Heuristics for Firefighter were introduced shortly after the problem itself was introduced and proved to be NP-complete. Metaheuristics are generally much more effective for FIREFIGHTER than the simple greedy algorithms that have been studied.

An early example of a heuristic is Hartnell and Li's greedy algorithm on trees [62] mentioned in Section 3.2.1. As long as one is willing to accept that the number of vertices saved by the algorithm is, at worst, half the optimal number saved, we have an efficient algorithm that approximately solves the optimization problem. Recent heuristics and metaheuristic approaches perform much better in practice. Garcia-Martinez, Blum, Rodriguez and Lozano [55] summarized a few of the existing elementary heuristics, and introduced a number of new ones.

Other heuristics for Firefighter have been studied. These are discussed in the following sections. Metaheuristics are generally much more effective for FIREFIGHTER than the simple greedy algorithms that have been studied.

3.5.1 Ant Colony Optimization

Ant colony optimization (ACO), is a class of optimization algorithms loosely based on the behaviour of ants in an ant colony. Artificial "ants" (agents) locate optimal solutions by running through a parameter space representing all possible solutions, then a set of superior solutions are identified, and new agents investigate solutions within the set of supposedly superior solutions, and decide which of the solutions of this set are the best. This is similar to the behaviour of ants in an ant colony as real ants lay down pheromones directing one another to resources while moving through their environment. The agents in the simulation mark specific sets of solutions as possible candidates for an optimal or close-to-optimal solution

[90].

Blum, Blesa, García-Martínez, Rodríguez, and Lozano [12] considered the sequence of defended vertices as a permutation of the vertices of the graph. In the pure ACO algorithm they presented, a pheromone model is used where superior solutions are given a pheromone value $\tau_{v,j}$ for each combination of a vertex v and a position j of a permutation. Then a set of solutions is probabilistically generated based on pheromone information and a greedy decision making algorithm. Afterwards the pheromone values of each permutation and vertex are modified using a set of three solutions, one of which is the best solution found at that point of the algorithm (best here refers to most vertices saved). Lastly the simulation runs again where more solutions are probabilistically generated, and the probability of a vertex being defended depends on the pheromone value of a specific vertex defended at a specific point in the simulation.

Blum et al. [12] also introduced a hybrid ant colony optimization (hyACO) where the pure ACO approach is combined with a linear programming model. First, the pure ACO is applied numerous times with a computational time limit, then the best solution to the application of the pure ACO is given to a CPLEX linear programming solver, where it undergoes solution polishing, which modifies the proposed solution with a search strategy. The best solution found after this phase is provided as the output.

For large, dense graphs, the authors concluded that the hybrid ant colony optimization outperformed a pure CPLEX solver in terms of finding better solutions to the classic firefighter problem (more vertices saved, on average) [12].

Hu, Windbichler and Raidl [63] investigated the use of a variable neighbourhood search (VNS) for Firefighter and compared those results to the results in [12]. They did not obtain appreciably better results than the hybrid ant colony optimization heuristic.

3.5.2 Evolutionary Algorithms and The Multi-Objective Firefighter Problem

An *evolutionary algorithm* (EA) is a metaheuristic optimization algorithm which mimics the natural selection and evolution found in the animal kingdom. In these metaheuristics, randomly generated candidate solutions take the position of a species' first generation. The solutions with the best results are considered suitable for procreating. Those individuals (candidate solutions) are then "bred" and produce offspring (further solutions) that share a combination of their parents' genes (qualities from their parent solutions), the poorer quality solutions are replaced by the superior offspring, and the process repeats. Within the breeding step, solutions undergo *crosssover* and *mutation*. Crossover is the combination of parent genes, while mutation is the random alteration of gene values from their initial state. This allows random changes to the characteristics of the solutions, which can result in the final output being much closer to optimal. The probability of mutations for any given solution must be low, however, because if it is too high then the evolutionary algorithm essentially becomes completely randomized.

Suppose that for each vertex of the graph v_i, there are m values $v_{i_j}(v), j = 1, 2, ..., m$ assigned to each vertex. Each v_{i_j} can be interpreted as the worth of a vertex with respect to different criteria, for example, financial value and cultural importance. The m objectives f_j correspond to the sum of the value of vertices that are defended or untouched at the conclusion of the firefighter process. The quality of solutions to the multi-objective firefighter problem is defined according to a numerical weighting system that relates the relative importance of each objective. Optimal solutions are ones where the weight of the saved vertices is maximized.

Michalak [82], used a set of 10 crossover operators and five mutation operators. In order to decide which crossover and mutation operators were the best for solving Firefighter, the paper used two auto-adaptation mechanisms. The first mechanism evaluated the number of times an operator produced an improved result for either of the two objectives introduced above, the second mechanism evaluated the ratio between the number of times an operator was used and the number of times doing so resulted in an improvement in either objective. The probability of an operator being used at a particular time step depended on its scores on the two mechanisms. The automatically adapting evolutionary algorithm was run for graphs with a various number of vertices and an edge density that was not so low that the fire was too easily contained, but not so high that all but the defended vertices were burned. The author found that in terms of crossover, the cycle, order based, and position based crossovers performed the best, and in terms of mutations the highest scores were given to the insertion mutation. More information on these crossover and mutation operators can be found in [47, 89].

Michalak [83] built on his previous paper and focused on the multi-objective firefighter problem, but used N_{sub} weight vectors $W_0 = \{\lambda^{(1)}, \lambda^{(2)}, ..., \lambda^{(N_{sub})}\}$ which determined the goal of the search function. If the number of possible objectives (financial value, cultural importance, etc.) is equal to m, then each vector $\lambda^{(j)} = [\lambda_1^{(j)}, \lambda_2^{(j)}, ..., \lambda_m^{(j)}]$ sets weights for each of the objectives between 1 and m. For example, the weight vector $\lambda = [1/3, 2/3]$ corresponds to the second objective being twice as important as the first objective with regards to an optimal solution. This algorithm works in a similar way to the evolutionary algorithm mentioned above, but in this case the specimens are divided into N_{sub} sub-populations, and each subpopulation (each consisting of N_{pop} specimens) has a different weighting vector for prioritizing the multiple objectives. The algorithm makes use of a migration mechanism whereby specimens that are a member of specific sub-populations can migrate to other sub-populations, where vectors with similar weighting schemes are more likely to have migrations between their specimens. After migrations take place, an autoadaptation mechanism is used which is similar to the one mentioned in his previous paper. Michalak concluded that the Sim-EA algorithm was more computationally efficient than its predecessors.

Michalak and Knowles [86] investigated a nondeterministic version of Firefighter, where the fire spreads from a burning vertex to an undefended adjacent vertex independently with probability $p < 1$. They used an evolutionary algorithm to compare various heuristics that dictate the placement of firefighters and found that for probability p, where $0.7 \leq p < 1$, the evolutionary algorithm was effective, but for lower probabiliies it was not appreciably better than existing heuristics.

An *estimation of distribution algorithm* (EDA), is similar to an evolutionary algorithm, with the exception that in an EA, genetic operators such as mutation and crossover are used, while in an EDA a probabilistic model is built from the existing population and used to generate subsequent generations of specimens. Michalak [85] used several different EDAs and compared their performance in terms of solution quality. It was shown that an EDA based on the relationship between the current state of the graph (in terms of burning, non-burning, and untouched vertices) and the individual placement of firefighters outperformed the other EDAs. In [84] Michalak also considered an EA but used a local search technique called ED-LS which used heuristics to reduce the size of the neighbourhoods in which solutions are sought, resulting in greater computational efficiency. The author concluded that the ED-LS method had better results than methods using only EAs, and required much fewer computational resources than methods with more exhaustive local search techniques. Lipinksi [76] used an EA based on an EDA to obtain solutions for a variant of the Firefighter entitled "FFP2", where there were also floods that spread amongst vertices, and a vertex could be burning, flooded, both, or neither, at a given time. Lipinski concluded that the EA found solutions for FFP2 that were close to the optimal solutions found by a systematic enumeration of candidate solutions, or at least close to the quasi-optimal solutions found by an exhaustive local search.

3.5.3 A Matheuristic

Ramos, de Souza, and de Rezende [94] developed the first *matheuristic* for Firefighter. A *matheuristic* is an optimization algorithm that consists of interoperation between metaheuristics and mathematical programming. Here, a greedy randomized algorithm searches for candidate solutions, then the candidate solutions are selected based on "quality" (which is the number of vertices saved) and "diversity" (in terms of computational cost), then the selected solutions undergo an adaptive neighbourhood size search strategy to perform a local search on each solution and return the best solution found. The last step involves a modified version of Develin and Hartke's ILP. The authors not only found that their modified ILP approach was twice as efficient as Develin and Hartke's ILP, but that their novel matheuristic, on average, performed better (saved more vertices) than existing metaheuristic approaches.

Chapter 4

The Surviving Rate of Graphs

4.1 Introduction

For finite graphs, the most common objective is to save the maximum number of vertices. A natural extension of this concept to infinite graphs is to either terminate the spreading process in a finite number of time steps (which saves all but a finite number of vertices), or to save the maximum possible proportion of the vertices. As we will see in what follows, the concept of *surviving rate*, which is the proportion of vertices that can be saved, averaged over all possible outbreaks of fire, has received substantial attention in the literature. We now discuss surviving rates in both finite and infinite graphs.

4.2 Surviving Rates

Finbow et al. [48] introduced the concept of *expected damage*, which is the expected proportion of burned vertices when the fire breaks out at a random vertex of a graph. (Also see [35].) The closely related concept of the *b-surviving rate* $\rho_b(G)$ of a graph G was defined by Cai and Wang [21] as the average proportion of vertices that can be saved by b firefighters at each time step when a single fire breaks out at a random vertex of the graph. Recall that $MVS(G, v)$ is the maximum number of vertices of G that can be saved by one firefighter when a fire breaks out at the vertex v. We now extend this concept to multiple firefighters. Let $MVS(G, v, b)$ be the maximum number of vertices of G that can be saved by b firefighters when the fire breaks out at the vertex v. Then, for a graph G of order n having vertex set V, the *b-surviving rate* is

$$\rho_b(G) = \sum_{v \in V} \frac{MVS(G,v,b)}{n^2}.$$

In some cases the outbreak is not limited to a single vertex. In this case we define the surviving rate for an outbreak of f fires, to be, for F such that $|F| = f$,

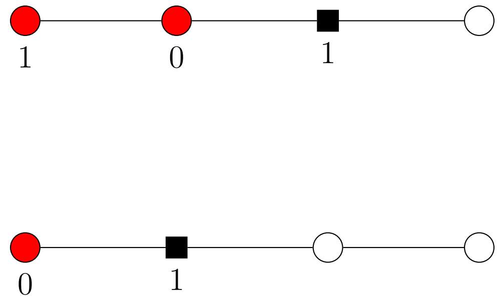

Figure 4.1: Consider $\mathcal{P}_4$. If the fire breaks out at an end vertex, then $\frac{3}{4}$ of the vertices can be saved, otherwise only $\frac{1}{2}$ can be saved. Since there are an equal amount of end vertices and interior vertices, the surviving rate is the average of these two values, $\frac{5}{8}$. The numbers correspond to the time step in which each vertex was defended or burned.

$$\rho_b(G, f) = \sum_{F \subseteq V} \frac{MVS(G,F,b)}{n\binom{n}{f}}.$$

For simplicity, $\rho_1(G)$ is usually written $\rho(G)$. For example, $\rho(C_n) = 1 - \frac{2}{n}$, as on any cycle with a single outbreak, one can save all but two vertices with one firefighter.

Wang, Finbow, and Wang [97] considered b-surviving rates in general graphs.

Theorem 4.2.1. [97] *For any $\epsilon > 0$, almost all graphs have b-surviving rates less than ϵ.*

For the objective of saving the maximum proportion of vertices, a class of graphs is called *b-optimal* if the proportion of vertices saved under an optimal strategy with b firefighters available per time step is approaches 1 as the number of vertices in the graph approaches infinity. Formally, a class of graphs $\mathcal{G}$ is called *b-optimal* if, for all $G \in \mathcal{G}$, $\lim_{n\to\infty} \rho_b(G) = 1$.

Cai and Wang [21] proved that trees are 1-optimal.

Since very few classes of graphs are 1-optimal or even b-optimal (see Theorem 4.2.1), rather than searching for graph classes that have surviving rates arbitrarily close to 1, we may search for graph classes that have positive surviving rates. A class of graphs is called *b-good* if its b-surviving rate is bounded below by a positive constant. That is, the proportion of vertices saved as n approaches infinity is positive. Formally, a class of graphs $\mathcal{G}$ is called *b-good* if, for all n-vertex graphs $G \in \mathcal{G}$, $\lim_{n\to\infty} \rho_b(G) > 0$.

Messinger [81] noted that, for given b, while every b-optimal class of graphs is b-good, there are numerous examples of b-good graph classes that are not b-optimal. As mentioned above, almost all graphs are not b-good for any fixed integer b.

The first authors to investigate asymptotic results were Wang and Moeller [87]. They proved that any finite rectangular grid G of any size is 1-good.

Theorem 4.2.2. [87] *For any positive integer n, $\rho(P_n \square P_n) \geq \frac{1}{4} - \frac{1}{4n^2}$.*

It is easy to see here that the proportion of vertices that can be saved differs greatly depending on the location of the outbreak of the fire. If the fire is located at the corner of the finite rectangular grid, then an optimal strategy would save all but a single row (or column) of vertices, leading to $n^2 - n$ vertices being saved. If the fire breaks out in the middle of the grid, one can save a wedge representing approximately 1/4 of the vertices, otherwise, one can save a larger wedge with about half the vertices.

Since the number of corner vertices on any finite rectangular grid is 4, regardless of the size of n, averaging the above-mentioned values to approximate $\rho(P_n \square P_n)$ implies that the rectangular grid is 1-good but not 1-optimal. Moreover, since the number of corner and edge vertices is $O(n)$ and the number of interior vertices is $O(n^2)$, one might expect the 1-surviving rate of the finite rectangular grid to be close to $\frac{1}{4}$. However, the situation turns out to be different, as shown by Gavenčiak, Kratochvíl, and Pralat.

Theorem 4.2.3. [56] *For any positive integer n, $\rho(P_n \square P_n) \geq \frac{5}{8}$.*

For the infinite Cartesian grid $\mathbb{Z} \square \mathbb{Z}$, however, $\rho(\mathbb{Z} \square \mathbb{Z}) = \frac{1}{4}$ [56].

The *cubic grid G_{hex}* is the infinite planar 3-regular grid whose faces are hexagons. Gavenčiak et al. investigated the surviving rate of cubic grids.

Theorem 4.2.4. [56] *If G_{hex} is the the cubic grid, then $\rho(G_{hex}) \geq \frac{2}{3}$.*

Cai, Cheng, Verbin and Zhou [19] considered the surviving rate of graphs with bounded treewidth. (See [39] for a definition of treewidth.)

Theorem 4.2.5. [19]

(a) For n-vertex outerplanar graphs G, $\lim_{n\to\infty} \rho(G) = 1$.

(b) For every n-vertex tree T, $\rho(T) \geq 1 - \Theta(\frac{\log n}{n})$.

(c) For every graph G of treewidth k, $\rho_k(G) = 1 - O(\frac{k^2 \log n}{n})$.

The authors also note that there are graphs of treewidth k that are not $(k-1)$-good. More results for outerplanar graphs which have treewidth 2 can be found in [101].

Costa et al. [29] extended the lower bound of Cai et al. on the surviving rate of n-vertex trees to cases where there are arbitrarily many initial fires.

Theorem 4.2.6. [29] *Let (T, F) be an f-rooted tree with n vertices and with maximum degree Δ. Then $\rho(T, f) \geq 1 - O(\frac{\log n}{n})$.*

Let $k \geq 0$ be an integer. A k-planar graph is a graph that can be drawn in the plane such that each edge may be crossed at most k times by other edges. (A 0-planar graph is a planar graph.) For example, K_6 is 1-planar. Kong and Zhang investigated the surviving rate of 1-planar graphs.

Theorem 4.2.7. [70] *For a 1-planar graph G, $\rho_6(G) > \frac{1}{163}$.*

The authors of [70] state that they believe their bound of $\frac{1}{163}$ is not the best possible, and leave as an open problem the question of whether or not 1-planar graphs are k-good for some $k \leq 5$.

There is a substantial amount of literature on the surviving rate of planar graphs. It should be noted that, because of the planarity of $K_{2,n}$, an arbitrary planar graph may not even be 1-good. Wang, Kong, and Zhang [99] proved the following:

Theorem 4.2.8. [99] *If G is a planar graph with $n \geq 2$ vertices and without 4-cycles, then $\rho_2(G) > \frac{1}{76}$.*

Kong, Wang, and Zhu investigated the relationship between the surviving rate and the minimum degree of planar graphs.

Theorem 4.2.9. [69] *If G is a planar graph with minimum degree δ, then*

1. $\rho_4(G) > \frac{3}{11}$ *if* $\delta = 5$,
2. $\rho_4(G) > \frac{3}{19}$ *if* $\delta = 4$,
3. $\rho_4(G) > \frac{1}{9}$ *if* $\delta \leq 3$.

The proofs use maximal planar graphs to obtain the lower bounds on the surviving rate.

It was mentioned in Proposition 2.2.1 that if H is a spanning subgraph of G, then for a given outbreak and number of firefighters, one can save at least as many vertices of H as they can vertices of G. This implies the next result.

Proposition 4.2.10. *If H is a spanning subgraph of G, then, for $b \geq 1$, $\rho_b(G) \leq \rho_b(H)$.*

Pralat [91] investigated the b-surviving rate of graphs with low average degree relative to the number of firefighters.

Theorem 4.2.11. [91] *For any $\epsilon > 0$ and $b \geq 2$, any graph G on n vertices with average degree at most $b + 2 - \frac{1}{b+2} - \epsilon$ has b-surviving rate $\rho_b(G) > \frac{2\epsilon}{5(b+2-\frac{1}{b+2})} > 0$.*

Pralat also constructed a family of random graphs $\mathcal{G}$ with average degree $b+2-\frac{1}{b+2}$ such that, for $G \in \mathcal{G}$, $\lim_{n\to\infty} \rho_b(G) = 0$, to show that the threshold maximum average degree $b+2-\frac{1}{b+2}-\epsilon$ cannot be improved.

When $b = 1$ we have $b + 2 - \frac{1}{b+2} = 2\frac{2}{3}$. Hence graphs G with average degree at most $2\frac{2}{3} - \epsilon$ have $\rho(g) > \frac{2\epsilon}{5\frac{8}{3}} = \frac{3\epsilon}{20}$. A related result with a slightly weaker bound on the average degree was proved in [92].

Theorem 4.2.12. [92] *Let G be a graph with $n \geq 2$ vertices and average degree at most $\frac{30}{11} - \epsilon$ for some $0 < \epsilon < \frac{1}{2}$. Then $\rho(G) \geq \frac{\epsilon}{30}$.*

Esperet, Van den Huevel, Maffray, and Sipma [45] proved that an arbitrary planar graph was 4-good (2-good for triangle-free planar graphs). This result was independently improved to show that every planar graph is 3-good [57, 72]. The question of whether all planar graphs are 2-good was posed by Wu, Kong, and Wang [102] and remains open. Though an arbitrary planar graph need not be 1-good, certain restricted types of planar graphs are 1-good.

Wang, Finbow, and Wang [98] proved that any planar graph G with girth at least 7 is 1-good. In [96] the same authors proved a lower bound on the 2-surviving rate of planar graphs without 6-cycles. Wu et al. [102] proved that any planar graph G without 5-cycles was 2-good. Wang, Wu, Hu, and Wang [100] strengthened this result to planar graphs without chordal 5-cycles. (A chordal n-cycle is a cycle of length n with at least one additional edge between non-consecutive vertices in the cycle.) Various authors have considered the relationship between the cycle lengths in a planar graph and the b-surviving rate. We conclude this chapter by summarizing their results.

Theorem 4.2.13. *(a)* [57] *For an arbitrary planar graph G,* $\rho_3(G) \geq \frac{2}{21}$.

(b) [45] *For any triangle-free planar graph G,* $\rho_2(G) > \frac{1}{723636}$.

(c) [98] *For any planar graph G with girth at least* 7, $\rho_2(G) > \frac{1}{301}$.

(d) [96] *For any planar graph G without* 6*-cycles,* $\rho_2(G) > \frac{1}{85}$.

(e) [102] *For any planar graph G without* 5*-cycles,* $\rho_2(G) > \frac{1}{363}$.

(f) [58] *For any planar graph G with average degree* $\frac{9}{2} - \epsilon$, *where* $\epsilon \in (0, 1]$, $\rho_2(G) \geq \frac{2}{9}\epsilon$, *which implies (g) and (h).*

(g) [58] *For any triangle-free planar graph G on at least two vertices,* $\rho_2(G) > \frac{1}{9}$.

(h) [58] *For any* 4*-cycle-free planar graph G on at least two vertices,* $\rho_2(G) > \frac{1}{9}$.

(i) [100] *For any planar graph G without chordal* 5*-cycles,* $\rho_2(G) > \frac{1}{1083}$.

Theorem 4.2.13 (h) improved the previous bound on the 2-surviving rate of planar graphs without 4-cycles of $\frac{1}{76}$ obtained in [99].

Chapter 5

More Variants on the Firefighter Problem

5.1 Introduction

Many variants of Firefighter have been studied in the literature. Each has their own set of surprising results and research directions. We explore a selection of variants in this chapter.

5.2 Spreading Vaccinations Model, Resource Minimization for Fire Containment, and Constrained Fires

Anshelevich, Chakrabarty, Hate, and Swamy [4] introduced a variant of Firefighter called the "spreading vaccinations model". This model was also investigated by Floderus, Lingas, and Persson [51]. In this variant of Firefighter, vertices that are defended spread their protection to neighbouring vertices at each time step in the same manner as the fire, that is, a vertex will be protected if it has a protected neighbour by or before the time step that it has an infected neighbour. Though this is not a realistic model of vaccine protection, if one considers the vertices of the graph to be people or households, the "fire" to be competing rumours or gossip, and the defended vertices considered to be people receiving an alternative, beneficial idea, then this model could be used to simulate the process as ideas, both good and bad, tend to spread between individuals and households. In this paper, the authors investigate Firefighter (which they call "MaxSave") for both the traditional and spreading vaccinations model, as well as another version which they call "MinBudget" (for both spreading and classic models). In MinBudget, which was studied by Chalermsoook and Chuzhoy [22] as RMFC, or "Resource Minimization for Fire Containment", there is a special set S of vertices which are to be protected, (similarly to S-Fire), and the goal is to find the minimum number of fixed firefighters per round that can protect all the vertices in S if employed under an optimal strategy. Floderus et al. proved that for the spreading vaccinations model, MinBudget is NP-complete, and for the classic model, they found a $2\sqrt{n}$-approximation for MinBudget on general graphs.

Chalermsook and Chuzhoy [22] provided an $O(\log^* n)$ approximation algorithm for RMFC on trees, along with a few other results, including an $O(\log n)$-approximation for RMFC on directed layered graphs. Lee [74] found that it is NP-hard to approximate RMFC within any constant factor if their variant of the unique games conjecture (UGC) is true. (The original UGC appeared in [65], whereas [74] explicitly defines the variant which is a sufficient condition for the NP-hardness of approximating RMFC.)

Bonato, Messinger and Pralat [13] introduced a variant of Firefighter with "constrained fires". In this version, called k-firefighter, the fire can only burn at most k unprotected vertices per round. They defined what we will call here the *k-constrained-1-surviving rate*, which is the 1-surviving rate of a graph that can only burn at most k vertices per round. They found that for the infinite random graph, the k-constrained-1-surviving rate is bounded below by $\frac{1}{k+1}$.

Alvarez, Blesa, and Molter [2], (also see [3]), investigated concepts called *Price of Anarchy* (PoA) and *Price of Stability* (PoS). Both involve a version of Firefighter where firefighters are placed by a different player in each round. Before we can define these terms, we must define what an equilibrium is. In this version we consider the game $\mathcal{G}^{(\text{Selfish})}$ where the objective for the individual is to maximize the number of vertices saved by the placement of their firefighter, and the game $\mathcal{G}^{(\text{Non-profit})}$, where the objective for the individual is for the game to end with the maximum number of vertices saved, regardless of which firefighter saves more vertices.

A strategy is a *(Nash) equilibrium* if each player knows the equilibrium strategies of the other players and no player can benefit by changing only their own strategy. A strategy that achieves a Nash equilibrium is called a Nash equilibrium strategy profile. Note that there can be multiple equilibria for an arbitrary game (including Firefighter), and it is not necessary for the optimal solution be a Nash equilibrium. A *social welfare* function $W(s)$ is a quantitative measure of a goal for a particular game that describes the collective success of players in a game, without regard to their individual successes. Here s is a strategy, S is the set of all strategies, and the social welfare function $W(s)$ refers to the number of vertices saved, i.e. $Max_{s \in S} W(s)$ denotes $MVS(G, r)$. Here, the price of anarchy is equal to the ratio between the optimal solution and the worst equilibrium in terms of the social welfare function $W(s)$. In general, for a game $\mathcal{G}$, $PoA(\mathcal{G}, W) = \frac{Max_{s \in S} W(s)}{Min_{s \in E} W(s)}$, where $E \subseteq S$ is the set of all Nash equilibrium strategy profiles.

The Price of Stability (PoS) delineates the ratio between the optimal solution overall and the best equilibrium in terms of the social welfare function $W(s)$. In general, for a game $\mathcal{G}$, $PoS(\mathcal{G}, W) = \frac{Max_{s \in S} W(s)}{Max_{s \in E} W(s)}$, where $E \subseteq S$ is the set of all Nash equilibrium strategy profiles.

The authors found, surprisingly, that the behaviour of the selfish firefighters resulted in the same equilibria as the behaviour of the non-profit firefighters. This implies the following results:

$$PoS(\mathcal{G}^{(\text{Selfish})}, W) = PoS(\mathcal{G}^{(\text{Non-profit})}, W), \text{ and}$$
$$PoA(\mathcal{G}^{(\text{Selfish})}, W) = PoA(\mathcal{G}^{(\text{Non-profit})}, W).$$

The authors found that while the PoA for general graphs is quite high, the PoA for trees is at most 2. The reason for this is as follows. Since the PoA is the same for the selfish and the non-profit firefighters, we will focus on the behaviour of the selfish firefighters. Selfish firefighters place firefighters in such a way that their move immediately saves the greatest number of vertices. This is equivalent to the strategy of saving a vertex with the largest possible subtree. Applying this decision-making algorithm at each step is Hartnell and Li's greedy algorithm that always saves at least half the maximum number of vertices. Therefore, it is clear that the Nash equlibria are exactly the strategies in which the firefighters are employing Hartnell and Li's greedy algorithm. It follows that that PoA for trees is at most 2.

The authors also investigated the strategy of "coalitions", which we will not explore here. An updated version of the paper entitled *Firefighting as a Strategic Game* [3] appeared in 2016.

Michalak and Knowles [86] introduced the concept of "online firefighting". In "online firefighting", the number of firefighters available at each time step varies, and is only known at the beginning of that round. This impacts the player's ability to strategize in terms of finding optimal solutions because there is an element of the unknown with regard to future rounds. This paper investigates the use of simheuristics to solve both the online and the offline versions; see Section 3.5 for the definition of a simheuristic.

Online firefighting was also studied by Coupechoux, Demange, Ellison, and Jouve [33] (also see [34]). This paper investigated the use of online and offline algorithms for Firefighter on trees. They found that Hartnell and Li's greedy algorithm was a $\frac{1}{2}$-approximation for online versions of Firefighter. The authors also found an online optimization algorithm that is a $\frac{1}{\phi}$-approximation (where ϕ is the golden ratio), as long as the number of firefighters available at each round is at most two. In addition, they found that for infinite rooted trees with linear growth (which means there is a constant c such that the number of vertices of distance i from the root r is at most ci), any sequence of firefighters with at least as many total firefighters used by any time step $i \geq 1$ as some non-zero periodic sequence of firefighters is enough to contain the fire, even when revealed online.

5.3 Directed Graphs

We note that all of the results for Firefighter on trees ($f = 1$) can be viewed as results on rooted oriented trees where all edges are oriented away from the root.

Biebighauser, Holte, and Wagner [11] investigated Firefighter for regular orientations of the Cartesian grid $\mathbb{Z}\Box\mathbb{Z}$. The authors found that a single fire in any infinite regular directed grid can either be contained by a single firefighter, or by a single firefighter with one additional firefighter at some time step.

Kong, Zhang and Wang [71] investigated the surviving rate for oriented graphs, and prove the following results. Here, a k-degenerate graph is a graph G such that every induced subgraph H of G contains a vertex of degree at most k in H. A k-degenerate oriented graph is any orientation of a k-degenerate graph[1].

Theorem 5.3.1. [71]

(a) If $\vec{G}$ is a k-degenerate oriented graph, then $\rho_k(\vec{G}) \geq \frac{1}{k+1}$.

(b) If $\vec{G}$ is a planar oriented graph, then $\rho_2(\vec{G}) > \frac{1}{40}$.

(c) If $\vec{G}$ is a planar oriented graph without 4-cycles, then $\rho_1(\vec{G}) > \frac{1}{51}$.

Bensmail and Brettell [10] explored a version of the firefighting game on digraphs, where the firefighters select an orientation of the graph before the fire breaks out, and then the fire breaks out at either a randomly or strategically selected vertex. When one firefighter is available at each time step, then for complete graphs or complete bipartite graphs, it is possible to describe optimal strategies. The authors also provide lower bounds on the number of vertices that can be saved as a function of the chromatic number, of the maximum degree, or of the treewidth of a graph.

MacGillivray and Redlin [77] prove that for any orientation of the cubic grid (as defined in Section 4), a single firefighter is sufficient to contain a single fire. This is in contrast to the result of Gavenciak et al. [56], that proved that on the undirected infinite cubic grid, one firefighter in each round plus a total of two additional firefighters at some (not necessarily distinct) time step(s) is sufficient to contain a single outbreak, and conjecture that one firefighter is not enough to contain a single outbreak.

In addition, Clarke, Finbow, Fitpatrick, Messenger and Nowakowski [77] investigated a variation on Firefighter called "Seepage" in which only one vertex at a time is burned, and k vertices are defended per round as in Firefighter. The problem of defending against the seepage essentially becomes one of protecting the "sinks" in the directed graph. In this context, the *green number* is the minimum number of defended vertices per round required to stop

[1] Here the authors use the terms "digraph" to refer to oriented graphs. (A digraph may have multiple edges between vertices while an oriented graph cannot.) All graphs studied in this paper are oriented graphs.

a sink from burning. Bonato, Mitsche, and Pralat investigated the green number of certain classes of directed acyclic graphs in [14].

5.4 Geometric Firefighting

When Firefighter is viewed as an attempt to model the spread of fire, one could argue that a continuous model may be more true to life.

Geometric Firefighting is a firefighting process in the Euclidean plane that is of interest to researchers in computational geometry. There is an outbreak of fire at the origin, and the fire spreads in a radial fashion from the outbreak as the time t increases. That is, unless the fire is obstructed, it spreads continuously in all directions from the origin at a given speed, forming a circle of fire of increasing radius centred at the origin. At the same time, the firefighters build a wall with some speed relative to that of the spreading of the fire. The fire spreads along straight lines, so once a line touches this wall the fire cannot spread further along it. There has been interest in the shape and speed at which the firewall must be built in order to stop a fire that spreads radially or in a particular direction determined by prevailing winds. Bressan, Burago, Friend, and Jou [16] found that a single continuous wall built to stop a single outbreak of fire must be in a spiral shape in order to contain the fire with the minimum area burned. Bressan and Wang [17], found that the fire could be entirely contained if and only if the speed of the construction of the wall was strictly greater than the speed of the propagation of the fire. There are a number of papers on this interesting subject, see [5, 6, 7, 15, 68, 67, 106, 107].

Chapter 6
Open problems

6.1 Introduction

The following sections contain a short list of the open problems, conjectures, and directions for future research that were listed in Finbow and MacGillivray's survey [50], and subsequent papers on these topics.

6.2 Problems

The references after each problem or conjecture refer to publications where the problem was investigated.

1. Study the function $\ell(G, F)$, the minimum number of time units needed to contain f fires that break out at vertices in F, where this quantity is defined to be infinity if the fires cannot be contained. (See [79, 80, 81].)

2. Find algorithms and complexity results for the natural extension of S-FIRE in which there are $d \geq 2$ firefighters. (See [9, 27, 41].)

3. Given a weighted graph (G, r) of maximum degree 3 and in which r has degree 2, is there a polynomial-time algorithm that finds the maximum weight subset of vertices that can be saved? (See [30, 42].)

4. Is there a constant $c \in (0, 1]$ such that the "greedy" strategy of defending the vertex of highest degree adjacent to a burning vertex saves at least $c \cdot \mathrm{MVS}(T, r)$ vertices of a rooted tree (T, r)? (See [41].)

5. For a subset $S \subseteq V(G) - \{r\}$ of vertices of a rooted graph (G, r), define $F_{(G,r)}(S)$ to be the minimum number of firefighters needed to save all vertices in S. Find an expression for $F_{(T,r)}(S)$, where S is the set of leaves of the tree T. Under what conditions is this number equal to $\Delta - 1$, where Δ is the maximum degree of T? More generally, study

$$\text{Max} \sum_{v \in V} d_v w(v)$$

Subject to:

$$\begin{cases} \displaystyle\sum_{\text{level(v) = i}} d_v \leq 1 & \text{for each level } i \\ \\ d_v + \displaystyle\sum_{u \text{ an ancestor of } v} d_u \leq 1 & \text{for every leaf } v \text{ of } T \\ \\ d_v \in \{0,1\} \end{cases}$$

Figure 6.1: A 0-1 integer program for Firefighter on a tree.

$F_{(G,r)}(S)$ and $F_{(T,r)}(S)$ especially in the case when $f = 1$. (See [1, 4, 22, 74].)

6. Investigate Firefighter on trees in the cases where there are more than one fire and more than one firefighter. Determine the the approximation ratio of the greedy algorithm on trees for an arbitrary number f of fires and d of firefighters. (See [9, 29, 64].)

7. Is there a constant c such that the optimum solution to the LP relaxation of the 0-1 integer program in Figure 6.1 is at most c times the optimum integral solution? That is, does linear programming give a c-approximation algorithm for Firefighter on trees? (See [1, 23].)

8. Find classes of trees for which Firefighter can be solved in polynomial time. Find a structural characterization of the trees for which the LP relaxation of the 0-1 integer program in Figure 6.1 gives an optimum solution, or identify non-trivial classes of such trees. (See [23].)

9. Alspach [103] suggested a heuristic for trees in which the LP relaxation of the 0-1 integer program in Figure 6.1 is solved, and then the resulting solution vector is used as a probability distribution for a randomized algorithm for defending the tree. Investigate the performance of such a heuristic experimentally and theoretically. (See [20].)

10. Does one firefighter suffice to contain a fire in the infinite hexagonal grid? What about any finite number of fires? (See [56].)

11. Consider the following theorem.

Theorem 6.2.1. *(a)* [81] *In the 2-dimensional infinite strong grid, four firefighters are necessary and sufficient to contain any finite number of fires.*

(b) [52, 81] *In the 2-dimensional infinite triangular grid, three firefighters are necessary and sufficient to contain any finite number of fires.*

(c) [81] *In the 2-dimensional infinite hexagonal grid, two firefighters suffice to contain any finite number of fires.*

Determine the minimum number of time units needed to contain a single fire, and the minimum number of vertices that must burn over all strategies that use a given number of firefighters. (See [37, 56, 79].)

12. For $n \geq 3$, find the number of fractional firefighters needed to contain a single fire in the n-dimensional square grid and the n-dimensional strong grid. (See [88].)

13. It was conjectured in [88] that there is no periodic sequence for which the average, over all time units to the end of the process, of the number of firefighters used is less than $\frac{3}{2}$ that can contain an outbreak of a single fire in the 2-dimensional grid. This was proven to be true for all values $b \leq \frac{3}{2}$ in [46].

14. Improve the upper bound on the maximum number of vertices that can be saved when a fire in $P_n \square P_n$ breaks out at vertex (r, c), where $1 \leq r \leq c \leq \lceil n/2 \rceil$. Does the strategy in Proposition 6.2.2 save the maximum number of vertices?

    Proposition 6.2.2. [87] *In $P_n \square P_n$, when the fire breaks out at vertex (r, c), where $1 \leq r \leq c \leq \lceil n/2 \rceil$, if the firefighter defends vertices in the order $(r+1, c), (r+1, c+1), (r+2, c-1), (r+2, c+2), (r+3, c-2), (r+3, c+3), \ldots, (r+c, 1), (r+c, 2c), (r+c, 2c+1), \ldots, (r+c, n)$, then $n(n-r) - (c-1)(n-c)$ vertices are saved.*

    See [56].

15. It was conjectured in [87] that for any vertex v of $P_n \square P_n \square P_n$,

$$\lim_{n \to \infty} \frac{MVS(P_n \square P_n \square P_n, v)}{n^3} = 0.$$

    This was proven to be true in [59].

16. Investigate optimal graphs (with respect to expected damage) in situations other than $f = b = 1$ and $f = b = 2$. In particular, what about $f = 1$ and $b > f$? (See [19, 21, 97].)

17. It was conjectured in [21] that there is a positive constant c such that every nontrivial planar graph G of maximum degree three satisfies $\rho(G) \geq c$. (See [69].)

18. [21] Determine the minimum b, if it exists, such that there is a positive constant c such that every planar graph G satisfies $\rho_b(G) \geq c$. The best known lower bound for b was discovered independently in [57, 72].

19. Improve the lower bounds of Cai and Wang listed in the following theorem.

    Theorem 6.2.3. [21]

$$\rho(G) > \begin{cases} 1 - \sqrt{2/n} & \textit{if } G \textit{ is a tree with } n \textit{ vertices} \\ 1/6 & \textit{if } G \textit{ is an outerplanar graph} \\ 3/10 & \textit{if } G \textit{ is a Halin graph with of order at least } 5. \end{cases}$$

    Improvements on these bounds were made in [101, 105].

20. [21] Is it true that, for outerplanar graphs G, $\lim_{n\to\infty} \rho(G) = 1$? What about for Halin graphs? (See [19, 101, 105].)

21. [21] Is it true that, for every n vertex tree T, $\rho(T) \geq 1 - \Theta(\frac{\log n}{n})$? (See [19].)

22. [21] Prove that it is NP-hard to determine $\rho(T)$ for a given tree T. Not pursued in the literature.

23. [21] Determine the approximation ration for the greedy algorithm for the surviving rate $\rho(T)$ of a tree T. Is it $1 - \Theta(\frac{\log n}{n})$? (See [19].)

24. Given integers $f \geq 1$ and $b \geq 1$, an mmd-graph is a graph such that the number of vertices burned under an optimal strategy using b firefighters and maximized over all outbreaks of cardinality f is minimized. Describe the structure of mmd-graphs and use it to determine whether there are other mmd-graphs besides the ones listed in [48]. (See [19].)

25. What are the mmd-graphs when $f = d = 2$ and the number of vertices is less than 10? (See [19].)

26. Directed graphs can be used to model situations where a fire can spread from x to y but not from y to x (say because of topography or wind direction). Investigate Firefighter for (weighted) digraphs. (See [10, 28, 71, 77].)

Bibliography

[1] D. Adjiashvili, A. Baggio and R. Zenklusen. Firefighting on trees beyond integrality gaps. *ACM Trans. Algorithms,* **15** (2018), 20:1–20:33.

[2] C. Àlvarez, M. Blesa and H. Molter. Firefighting as a game. A. Bonato, F. Graham and P. Prałat (eds.), *Algorithms and Models for the Web Graph*, Springer International Publishing, 2014, 108–119.

[3] C. Àlvarez, H. Molter and M. Blesa. Firefighting as a strategic game. *Internet Mathematics,* **12** (2016), 101–120.

[4] E. Anshelevich, D. Chakrabarty, A. Hate and C. Swamy. Approximation algorithms for the firefighter problem: cuts over time and submodularity. Y. Dong, D. Du and O. Ibarra (eds.), *Algorithms Comput.*, Springer Berlin Heidelberg, 2009, 974–983.

[5] A. Barghi. *Firefighting on Geometric Graphs,* Ph.D. Dissertation, Dartmouth College, 2011.

[6] A. Barghi and P. Winkler. Firefighting on a random geometric graph. *Random Struct. Algorithms,* **46** (2015), 466–477.

[7] A. Barghi and P. Winkler. Firefighting on geometric graphs with density bounds. *Ars Combin.,* **131** (2017), 63–86.

[8] C. Bazgan, M. Chopin and M. Fellows. Parameterized complexity of the firefighter problem. T. Asano, S. Nakano, Y. Okamoto and O. Watanabe (eds.), *Algorithms Comput.*, Springer Berlin Heidelberg, 2011, 643–652.

[9] C. Bazgan, M. Chopin and B. Ries. The firefighter problem with more than one firefighter on trees. *Discrete Appl. Math.,* **161** (2013), 899–908.

[10] J. Bensmail and N. Brettell. Orienting edges to fight fire in graphs. *Australas. J. Combin.,* **71** (2018), 12–42.

[11] D. Biebighauser, L. Holte and R. Wagner. The firefighter problem for regular infinite directed grids. *Involve,* **5** (2012), 393–409.

[12] C. Blum, M. J. Blesa, C. García-Martínez, F. J. Rodríguez and M. Lozano. The firefighter problem: application of hybrid ant colony optimization algorithms. C. Blum and G. Ochoa (eds.), *Evol. Comput. Comb. Optim.*, Springer Berlin Heidelberg, 2014, 218–229.

[13] A. Bonato, M. Messinger and P. Prałat. Fighting constrained fires in graphs. *Theor. Comput. Sci.,* **434** (2012), 11–22.

[14] A. Bonato, D. Mitsche and P. Pralat. Vertex-pursuit in random directed acyclic graphs. *SIAM J. Discrete Math.,* **27** (2013), 732–756.

[15] A. Bressan. Differential inclusions and the control of forest fires. *J. Differ. Equ.,* **243** (2007), 179–207.

[16] A. Bressan, M. Burago, A. Friend and J. Jou. Blocking strategies for a fire control problem. *Anal. Appl.,* **06** (2008), 229–246.

[17] A. Bressan and T. Wang. The minimum speed for a blocking problem on the half plane. *J. Math. Anal. Appl.,* **356** (2009), 133–144.

[18] L. Cai, S. M. Chan and S. O. Chan. Random separation: a new method for solving fixed-cardinality optimization problems. *Lecture Notes in Comput. Sci.,* vol. 4169, 2006, 239–250.

[19] L. Cai, Y. Cheng, E. Verbin and Y. Zhou. Surviving rates of graphs with bounded treewidth for the firefighter problem. *SIAM J. Discrete Math.,* **24** (2010), 1322–1335.

[20] L. Cai, E. Verbin and L. Yang. Firefighting on trees: $(1-1/e)$-approximation, fixed parameter tractability and a subexponential algorithm. *Lecture Notes in Comput. Sci.,* vol. 5369, 2008, 258–269.

[21] L. Cai and W. Wang. The surviving rate of a graph for the firefighter problem. *SIAM J. Discrete Math.,* **23** (2009/10), 1814–1826.

[22] P. Chalermsook and J. Chuzhoy. Resource minimization for fire containment. *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms,* SIAM, 2010, 1334–1349.

[23] P. Chalermsook and D. Vaz. New integrality gap results for the firefighters problem on trees. K. Jansen and M. Mastrolilli (eds.), *Approx. Online Algorithms*, Springer International Publishing, 2017, 65–77.

[24] X. Chen, X. Hu, C. Wang and Y. Zhang. Continuous firefighting on infinite square grids. T.V. Gopal, Gerhard Jäger and Silvia Steila (eds.), *Theory Appl. Models Comput.*, Springer International Publishing, 2017, 158–171.

[25] J. Chlebíková and M. Chopin. The firefighter problem: a structural analysis. M. Cy-

gan and P. Heggernes (eds.), *Parameterized Exact Comput.*, Springer International Publishing, 2014, 172–183.

[26] J. Chlebíková and M. Chopin. The firefighter problem: further steps in understanding its complexity. *Theor. Comput. Sci.,* **676** (2017), 42–51.

[27] J. Choudhari, A. Dasgupta, N. Misra and M. S. Ramanujan. Saving critical nodes with firefighters is FPT. *Leibniz Int. Proc. Inform.,* vol. 80, 2017, Article 135.

[28] N. E. Clarke, S. Finbow, S. L. Fitzpatrick, M. E. Messenger and R. J. Nowakowski. Seepage in directed acyclic graphs. *Australas. J. Combin.,* **43** (2009), 91–102.

[29] V. Costa, S. Dantas, M. C. Dourado, L. Penso and D. Rautenbach. More fires and more fighters. *Discrete Appl. Math.,* **161** (2013), 2410–2419.

[30] V. Costa, S. Dantas, M. C. Dourado, L. Penso and D. Rautenbach. Slash and burn on graphs — firefighting with general weights. *Discrete Appl. Math.,* **210** (2016), 4–13.

[31] V. Costa, S. Dantas and D. Rautenbach. Asymptotic surviving rate of trees with multiple fire sources. *Discrete Appl. Math.,* **184** (2015), 14–19.

[32] V. Costa, S. Dantas and D. Rautenbach. The burning of the snark. *Mat. Contemp.,* **44** (2016), 1–10.

[33] P. Coupechoux, M. Demange, D. Ellison and B. Jouve. Online firefighting on trees. J. Lee, G. Rinaldi and A. Mahjoub (eds.), *Comb. Optim.*, Springer International Publishing, 2018, 121–132.

[34] P. Coupechoux, M. Demange, D. Ellison and B. Jouve. Firefighting on trees. *Theor. Comput. Sci.,* **794** (2019), 69–84.

[35] S. Crosby, A. Finbow, B. Hartnell, R. Moussi, K. Patterson and D. Wattar. Designing fire resistant graphs. *Congr. Numer.,* **173** (2005), 207–222.

[36] B. Das, M. K. Enduri, M. Kiyomi, N. Misra, Y. Otachi, I. V. Reddy and S. Yoshimura. On structural parameterizations of firefighting. *Theor. Comput. Sci.,* **782** (2019), 79–90.

[37] A. Dean, S. English, T. Huang, R. A. Krueger, A. Lee, M. Mizrahi and C. Wheaton-Werle. Firefighting on the hexagonal grid and on infinite trees. *arXiv:2010.05060v2,* (2020).

[38] M. Develin and S. G. Hartke. Fire containment in grids of dimension three and higher. *Discrete Appl. Math.,* **155** (2007), 2257–2268.

[39] Reinhard Diestel. *Graph Theory*, Springer, Berlin, 2018.

[40] R. Downey and M. Fellows. *Parameterized Complexity*, Springer, New York, 1999.

[41] C. Duffy. *A Collection of Algorithmic and Complexity Results for Variants of the Firefighter Problem,* Master's Thesis, University of Victoria, 2011.

[42] C. Duffy and G. MacGillivray. An analysis of the weighted firefighter problem. *J. Combin. Math. Combin. Comput.,* **94** (2015), 167–175.

[43] C. Duffy and G. MacGillivray. The firefighter problem: saving sets of vertices on cubic graphs. *Networks,* **74** (2019), 62–69.

[44] D. Dyer, E. Martínez-Pedroza and B. Thorne. The coarse geometry of Hartnell's firefighter problem on infinite graphs. *Discrete Math.,* **340** (2017), 935–950.

[45] L. Esperet, J. van den Heuvel, F. Maffray and F. Sipma. Fire containment in planar graphs. *J. Graph Theory,* **73** (2013), 267–279.

[46] O. Feldheim and R. Hod. 3/2 firefighters are not enough. *Discrete Appl. Math.,* **161** (2013), 301–306.

[47] A. Ferrolho and M. Crisóstomo. *Scheduling Jobs with Genetic Algorithms,* 197–207. Springer, Dordrecht, 2009.

[48] S. Finbow, B. Hartnell, Q. Li and K. Schmeisser. On minimizing the effects of fire or a virus on a network. *J. Comb. Math. Comb. Comput.,* **33** (2000), 311–322.

[49] S. Finbow, A. King, G. MacGillivray and R. Rizzi. The firefighter problem for graphs of maximum degree three. *Discrete Math.,* **307** (2007), 2094–2105.

[50] S. Finbow and G. MacGillivray. The firefighter problem: a survey of results, directions and questions. *Australas. J. Comb.,* **43** (2009), 57–77.

[51] P. Floderus, A. Lingas and M. Persson. Towards more effiicient infection and fire fighting. *Int. J. Found. Comput. Sci.,* **24** (2013), 3–14.

[52] P. Fogarty. *Catching the Fire on Grids,* Master's Thesis, University of Vermont, 2003.

[53] F. Fomin, P. Heggernes and E. van Leeuwen. Making life easier for firefighters. Springer Berlin Heidelberg, 2012, 177–188.

[54] F. Fomin, P. Heggernes and E. van Leeuwen. The firefighter problem on graph classes. *Theor. Comput. Sci.,* **613** (2016), 38–50.

[55] C. García-Martínez, C. Blum, F.J. Rodriguez and M. Lozano. The firefighter problem: empirical results on random graphs. *Comput. Oper. Res.,* **60** (2015), 55–66.

[56] T. Gavenčiak, J. Kratochvíl and P. Prałat. Firefighting on square, hexagonal, and triangular grids. *Discrete Math.,* **337** (2014), 142–155.

[57] P. Gordinowicz. Planar graph is on fire. *Theor. Comput. Sci.,* **593** (2015), 160–164.

[58] P. Gordinowicz. The 2-surviving rate of planar graphs with average degree lower than $\frac{9}{2}$. *J. Graph Theory,* **89** (2018), 341–349.

[59] S. G. Hartke. *Graph-theoretic Models of Spread and Competition,* Ph.D Dissertation, Rutgers, 2004.

[60] S. G. Hartke. Attempting to narrow the integrality gap for the firefighter problem on trees. *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.* vol. 70, 2006, 225–231.

[61] B. Hartnell. Firefighter! An application of domination. Presentation. *The 25th Manitoba Conference on Combinatorial Mathematics and Computing,* University of Manitoba, Winnipeg, Canada, 1995.

[62] B. Hartnell and Q. Li. Firefighting on trees: How bad is the greedy algorithm?. *Congressus Numerantium,* **145** (2000), 187–192.

[63] B. Hu, A. Windbichler and G. Raidl. A new solution representation for the firefighter problem. G. Ochoa and F. Chicano (eds.), *Evolutionary Computation in Combinatorial Optimization*, Springer International Publishing, 2015, 25–35.

[64] Y. Iwaikawa, N. Kamiyama and T. Matsui. Improved approximation algorithms for firefighter problem on trees. *IEICE Trans. Inf. Syst.,* **94-D** (2011), 196–199.

[65] S. Khot. On the power of unique 2-prover 1-round games. *Proceedings of the thirty-fourth annual symposium on theory of computing,* ACM, New York, 2002, 767–775.

[66] A. King and G. MacGillivray. The firefighter problem for cubic graphs. *Discrete Math.,* **310** (2010), 614–621.

[67] R. Klein, E. Langetepe and C. Levcopoulos. A fire fighter's problem. *31st International Symposium on Computational Geometry (SoCG 2015),* **34** (2015), 768–780.

[68] R. Klein, C. Levcopoulos and A. Lingas. Approximation algorithms for the geometric firefighter and budget fence problems. *Algorithms,* **11** (2018), Paper No. 45.

[69] J. Kong, W. Wang and X. Zhu. The surviving rate of planar graphs. *Theor. Comput. Sci.,* **416** (2012), 65–70.

[70] J. Kong and L. Zhang. A note on the surviving rate of 1-planar graphs. *Discrete Math.,* **340** (2017), 1074–1079.

[71] J. Kong, L. Zhang and W. Wang. The surviving rate of digraphs. *Discrete Math.,* **334** (2014), 13–19.

[72] J. Kong and W. Zhang, L.and Wang. Structural properties and surviving rate of planar graphs. *Discrete Math. Algorithms Appl.,* **06** (2014), Article 1450052.

[73] J.. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman

problem. *Proc. Amer. Math. Soc.,* **7** (1956), 48–50.

[74] E. Lee. Improved hardness for cut, interdiction, and firefighter problem. *44th International Colloquium on Automata, Languages, and Programming,* vol. 80, Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2017, Article 92.

[75] M. Leung. Fixed parameter tractable algorithm for firefighting problem. *arXiv:1104.1044,* (2011).

[76] P. Lipinski. Solving the firefighter problem with two elements using a multi-modal estimation of distribution algorithm. *2017 IEEE Congress on Evolutionary Computation (CEC),* (2017), 2161–2168.

[77] G. MacGillivray and S. Redlin. The firefighter problem on orientations of the cubic grid. *Bull. Inst. Combin. Appl.,* **88** (2020), 22–29.

[78] G. MacGillivray and P. Wang. On the firefighter problem. *J. Combin. Math. Combin. Comput.,* **47** (2003), 83–96.

[79] M. E. Messinger. Average firefighting on infinite grids. *Australas. J. Combin.,* **41** (2008), 15–28.

[80] Margaret-Ellen Messinger. Firefighting on the triangular grid. *J. Combin. Math. Combin. Comput.,* **63** (2007), 37–45.

[81] M.E. Messinger. *Firefighting on Infinite Grids,* Master's Thesis, Dalhousie University, 2004.

[82] K. Michalak. Auto-adaptation of genetic operators for multi-objective optimization in the firefighter problem. *International Conference on Intelligent Data Engineering and Automated Learning,* Springer, 2014, 484–491.

[83] K. Michalak. The sim-EA algorithm with operator autoadaptation for the multiobjective firefighter problem. G. Ochoa and F. Chicano (eds.), *Evolutionary Computation in Combinatorial Optimization*, Springer International Publishing, 2015, 184–196.

[84] K. Michalak. ED-LS – A heuristic local search for the multiobjective firefighter problem. *Appl. Soft Comput.,* **59** (2017), 389–404.

[85] K. Michalak. Estimation of distribution algorithms for the firefighter problem. B. Hu and M. López-Ibáñez (eds.), *Evolutionary Computation in Combinatorial Optimization*, Springer International Publishing, 2017, 108–123.

[86] K. Michalak and J. Knowles. Simheuristics for the multiobjective nondeterministic firefighter problem in a time-constrained setting. G. Squillero and P. Burelli (eds.), *Applications of Evolutionary Computation*, Springer International Publishing, 2016, 248–265.

[87] S. Moeller and P. Wang. Fire control on graphs. *J. Combin. Math. Combin. Comput.,* **41** (2002), 19–34.

[88] K.L. Ng and P. Raff. A generalization of the firefighter problem on $\mathbb{Z} \times \mathbb{Z}$. *Discrete Appl. Math.,* **156** (2008), 730–745.

[89] I. M. Oliver, D. J. Smith and J. R. C. Holland. A study of permutation crossover operators on the traveling salesman problem. *Proceedings of the Second International Conference on Genetic Algorithms on Genetic Algorithms and Their Applications,* L. Erlbaum Associates Inc., 1987, 224–230.

[90] S. Parsons. Ant colony optimization. *Knowledge Eng. Review,* **20** (2005), 92–93.

[91] P Prałat. Sparse graphs are not flammable. *SIAM J. Discrete Math.,* **27** (2013), 2157–2166.

[92] P. Prałat. Graphs with average degree smaller than $\frac{30}{11}$ burn slowly. *Graphs Comb.,* **30** (2014), 455–470.

[93] R. C. Prim. Shortest connection networks and some generalizations. *Bell Syst. Tech. J.,* **36** (1957), 1389–1401.

[94] N. Ramos, C. de Souza and P. de Rezende. A matheuristic for the firefighter problem on graphs. *Int. Trans. Oper. Res.,* **27** (2020), 739–766.

[95] A. Scott, U. Stege and N. Zeh. Politician's firefighting. *Lecture Notes in Comput. Sci.,* vol. 4288, 2006, 608–617.

[96] W. Wang, S. Finbow and J. Kong. The 2-surviving rate of planar graphs without 6-cycles. *Theor. Comput. Sci.,* **518** (2014), 22–31.

[97] W. Wang, S. Finbow and P. Wang. The surviving rate of an infected network. *Theor. Comput. Sci.,* **411** (2010), 3651–3660.

[98] W. Wang, S. Finbow and P. Wang. A lower bound of the surviving rate of a planar graph with girth at least seven. *J. Comb. Optim.,* **27** (2014), 621–642.

[99] W. Wang, J. Kong and L. Zhang. The 2-surviving rate of planar graphs without 4-cycles. *Theor. Comput. Sci.,* **457** (2012), 158–165.

[100] W. Wang, T. Wu, X. Hu and Y. Wang. Planar graphs without chordal 5-cycles are 2-good. *J. Comb. Optim.,* **35** (2018), 980–996.

[101] W. Wang, X. Yue and X. Zhu. The surviving rate of an outerplanar graph for the firefighter problem. *Theor. Comput. Sci.,* **412** (2011), 913–921.

[102] T. Wu, J. Kong and W. Wang. The 2-surviving rate of planar graphs without 5-cycles. *J. Comb. Optim.,* **31** (2016), 1479–1492.

[103] B. Yang, D. Dyer and B. Alspach. Sweeping graphs with large clique number. *Discrete Math.,* **309** (2009), 5770–5780.

[104] L. Yang. *Efficient Algorithms on Trees,* Ph.D. Dissertation, Chinese University of Hong Kong, 2009.

[105] X. Yue and W. Wang. The surviving rate of Halin graphs. *J. Zhejiang Norm. Univer. v34,* (2011), 141–144.

[106] M. Zambon, P. de Rezende and C. de Souza. Finding exact solutions for the Geometric Firefighter Problem in practice. *Comput. Oper. Res.,* **97** (2018), 72–83.

[107] M. Zambon, P. de Rezende and C. de Souza. Solving the geometric firefighter routing problem via integer programming. *Eur. J. Oper. Res.,* **274** (2019), 1090–1101.