







Article

The Moving Firefighter Problem

Bruno R. Gutiérrez-De-La-Paz ^{1,†}, Jesús García-Díaz ^{2,3,†}, Rolando Menchaca-Méndez ^{1,*,†},
Mauro A. Montenegro-Meza ^{1,†}, Ricardo Menchaca-Méndez ^{1,†} and Omar A. Gutiérrez-De-La-Paz ^{1,†}

¹ Centro de Investigación en Computación del Instituto Politécnico Nacional, Mexico City 07738, Mexico

² Consejo Nacional de Ciencia y Tecnología, Mexico City 03940, Mexico

³ Instituto Nacional de Astrofísica, Óptica y Electrónica, Coordinación de Ciencias Computacionales, Puebla 72840, Mexico

* Correspondence: rmen@cic.ipn.mx

† These authors contributed equally to this work.

Abstract: The original formulation of the firefighter problem defines a discrete-time process where a fire starts at a designated subset of the vertices of a graph G . At each subsequent discrete time unit, the fire propagates from each burnt vertex to all of its neighbors unless they are defended by a firefighter that can move between any pair of vertices in a single time unit. Once a vertex is burnt or defended, it remains in that state, and the process terminates when the fire can no longer spread. In this work, we present the moving firefighter problem, which is a generalization of the firefighter problem where the time it takes a firefighter to move from a vertex u to defend vertex v is determined by a function τ . This new formulation models situations such as a wildfire or a flood, where firefighters have to physically move from their current position to the location of an entity they intend to defend. It also incorporates the notion that entities modeled by the vertices are not necessarily instantaneously defended upon the arrival of a firefighter. We present a mixed-integer quadratically constrained program (MIQCP) for the optimization version of the moving firefighter problem that minimizes the number of burnt vertices for the case of general finite graphs, an arbitrary set $F \subset V$ of vertices where the fire breaks out, a single firefighter, and metric time functions τ .

Keywords: firefighter problem; np-hard; mathematical programming; mixed-integer quadratically constrained programming (MIQCP); exact algorithm; spread and containment in networks

MSC: 90-10; 90C35; 90C27; 90C11



Citation: Gutiérrez-De-La-Paz, B.R.; García-Díaz, J.; Menchaca-Méndez, R.; Montenegro-Meza, M.A.; Menchaca-Méndez, R.; Gutiérrez-De-La-Paz, O.A. The Moving Firefighter Problem. *Mathematics* **2023**, *11*, 179. <https://doi.org/10.3390/math11010179>

Academic Editors: Shi Qiang Liu, Erhan Kozan, Felix T. S. Chan, Weidong Li and Ioannis G. Tsoulos

Received: 16 November 2022

Revised: 15 December 2022

Accepted: 23 December 2022

Published: 29 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The firefighter problem was originally formulated by Bert Hartnell in 1995 [1]. It defines a discrete-time model of a diffusive process (e.g., a fire, a flood, an infectious disease, information, a computer virus, or an invasive species) on a graph $G = (V, E)$, where a fire breaks out at a set of vertices $F \subset V$. At each subsequent discrete time unit, the fire propagates from each burnt vertex to all of its neighbors unless they are defended by a firefighter. Once a vertex is burnt or defended, it remains in that state, and the process terminates when the fire can no longer spread. A solution to the problem defines the defending actions that have to be taken at each discrete time unit to optimally contain the spreading process by minimizing the number of burnt vertices or the time it takes to stop the diffusive process.

Over the years, the firefighter problem has gained growing relevance for its theoretical properties but also because it provides a simple model that can be used to study impactful phenomena such as the spreading and containment dynamics of viruses in social networks [2] and fires in oil terminals [3] and high-rise buildings [4].

Unfortunately, for the case of fighting actual fire outbreaks, the original formulation of the firefighter problem and its many variants and generalizations (e.g., [5–18]) fail to

capture an important aspect that arises in more realistic scenarios, namely, that the process of transporting (or moving) a firefighter from her current location to defend an entity located at a different location may take a variable amount of time. In general, this time may be a function of the distance between the two locations and the particular properties of the terrain, the entity to be defended, and the firefighter herself. As a concrete example, when fighting a wildfire, any strategy has to consider the time it takes to move fire engines and perhaps heavy machinery to a given region and then defend that region by creating control lines containing no combustible material. This time may depend on many factors, such as the land topography and the type of fire engines. It is worth noting that designing efficient wildfire suppression strategies has become increasingly important. During the 2019/2020 bushfire season, the number of hectares burned in Australia surpassed 6.75 million [19], and 4.09 million hectares burned in the United States in 2020 [20]. In 2021, more than 75 thousand wildfire outbreaks were detected in the Amazon Rainforest [21].

In this work, we introduce the moving firefighter problem, which is a generalization of the firefighter problem that incorporates a function τ that determines the time it takes a firefighter to move from its current location at vertex $u \in V$ to defend another vertex $v \in V$. Similarly to the original formulation of the firefighter problem, in this generalization, we assume that the time it takes the fire to spread from a burnt vertex to its neighbors in the graph is constant.

We present a mixed-integer quadratically constrained program (MIQCP) for the optimized version of the moving firefighter problem that minimizes the number of burnt vertices for the case of general finite graphs, an arbitrary set $F \subset V$ of vertices where the fire breaks out, a single firefighter, and metric time functions τ . We characterized the performance of the proposed solution through experiments on random graphs.

The rest of the paper is organized as follows. Section 2 presents a review of the extensive body of work on the firefighter problem and of its generalization and variants. Section 3 presents the formulation of the moving firefighter problem and the concepts and notation that are used throughout the paper. In this section, we also establish that the moving firefighter problem is indeed a generalization of the original firefighter problem, and hence, that it belongs to the class of \mathcal{NP} -hard problems. Section 4 presents the proposed exact solution. Section 5 presents the results of a series of experiments, and Section 6 presents our concluding remarks and future work.

2. Related Work

The firefighter problem has been extensively studied. It has been determined that it is \mathcal{NP} -hard for bipartite graphs [22], for cubic graphs [9], for trees of maximum degree three [22] and of pathwidth three [23], for co-bipartite graphs [23], and for unit disk graphs [24]. It has also been established that the problem is in P for caterpillars and P-trees [22], interval graphs, split graphs, permutation graphs, P_k -free graphs for fixed k [24], and if the fire breaks out at a vertex of degree at most two, for graphs of maximum degree three [25].

For the case of $b \geq 2$ firefighters, the problem is \mathcal{NP} -complete for trees of maximum degree $b + 2$ and \mathcal{NP} -hard for the optimization version that maximizes the number of saved (unburnt) vertices for trees of maximum degree $b + 3$ [11].

Regarding approximations, for the case of trees and a single firefighter, the greedy algorithm that every time selects the vertex that maximizes the number of protected vertices is a tight $1/2$ -approximation [2,26]. There is also a $(1 - 1/e)$ -approximation algorithm for rooted trees based on an LP-relaxation and randomized rounding [27]. The previous algorithm was used by Iwaikawa et al. to provide a 0.7144-approximation algorithm for ternary trees [28]. The problem is not $n^{1-\epsilon}$ -approximable on general graphs for any $\epsilon \in (0, 1)$ unless $P = NP$ [8]. More recently, Adjashvili et al. [29] presented a polynomial-time approximation scheme (PTAS) for trees.

From a parameterized point of view, the problem is $W[1]$ -hard on general graphs for $b \geq 1$ firefighters when parameterized by the number of saved vertices, protected vertices, and burned vertices [30].

Other generalizations of the firefighter problem include a version where the number of available firefighters at each time is not constant [7], a version where the objective is to protect a particular subset of the set of vertices $S \subseteq V$ [9,11], a version where the protection also propagates from each defended vertex to their non-burning neighbors [8,13], and a non-deterministic version where at each time the fire propagates to unprotected vertices with a given probability [14]. There is also a fractional version where vertices can be partially burned and protected, and partially burned vertices can propagate their fractions of the fire to their undefended or partially defended neighbors [5]. More recently, Coupechoux et al. introduced online versions of the firefighter and fractional firefighter problems where the number of available firefighters is revealed at each discrete time [16]. The k -firefighter problem is another generalization defined in terms of a two-player game where the fire propagates to at most k vertices in each round with the objective of burning as many vertices as possible while the firefighter defends vertices to contain the spread of the fire [6,10].

The geometric firefighter problem [12,15,17] is the closest variation to the moving firefighter problem because it considers the speed with which barriers are constructed to contain the fire. In this variation, the fire starts at a point inside a simple polygon \mathcal{P} and circularly propagates at a constant speed. A firefighter sequentially builds a series of one-dimensional barriers at a constant speed, which can be different from the fire propagation speed, to contain the propagation of the fire. Barriers have to be built continuously before the fire reaches any of its points. The objective is to maximize the protected area of \mathcal{P} , namely, the area separated from the fire by the barriers. Similarly to the original firefighter problem, the geometric firefighter problem assumes that the firefighter can be instantaneously transported from the last position of a recently constructed barrier to the origin of the next barrier in the solution.

In the traveling firefighter problem, also referred to as the L_2 -TSP (traveling salesman problem), a firefighter has to extinguish a set of wildfires located at positions modeled by vertices. The distance between vertices is defined by a function $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$. A solution to this problem is a permutation of the vertices that minimize the total damage produced by the fires, which is a quadratic function of time elapsed from when the fires broke out to when the firefighter arrives [18].

Some other \mathcal{NP} -hard problems that resemble the firefighter problem are the graph burning problem (GBP) [31–33] and the influence maximization problem (IMP) [34–36]. However, while they seek the rapid diffusion of information, the firefighter problem seeks diffusion containment. Nevertheless, very recently, the authors of [37] used the integer linear programming formulation (ILP) of the firefighter problem presented in [6] to derive the first exact solution to the graph burning problem.

Table 1 summarizes the main results discussed in this section.

Table 1. Summary of related work.

Reference	Contributions
[1]	Hartnell introduces the Firefighter Problem (FP).
[9,11,22–24]	The FP is NP-hard for different graph families.
[22,24,25]	Instances where the FP is in P.
[8,26–29]	Approximation results for the FP.
[30]	The FP is W[1]-hard on general graphs with more than one firefighter.
[6]	Presents an integer linear program (ILP) for the FP.
[7]	Presents a version of the FP with a variable number of firefighters.
[9,11]	Present the <i>S</i> -Fire version of the FP, which aims to protect a subset of the vertices.
[8,13]	A version of the FP where the protection propagates from defended nodes.
[14]	Version of the FP with nondeterministic fire propagation.
[5]	Introduces the Fractional Firefighter Problem.
[16]	An Online version of the Fractional Firefighter Problem.
[6,10]	Examine a two-player version of the FP called the <i>k</i> -Firefighter.
[12,15,17]	The Geometric Firefighter Problem.
[18]	Presents the Traveling Firefighter Problem (<i>L</i> ₂ -TSP)
[31–33]	Investigate the relation between the FP and the Graph Burning Problem.

3. The Moving Firefighter Problem

An instance of the moving firefighter problem (*MFP*) is defined as a 6-tuple $\langle G, F, a, \tau, f, T_S \rangle$, where $G = (V, E)$ is a simple graph; $F \subset V$ is a subset of the vertices where the fire breaks out; a is the depot where the firefighters are initially stationed (also referred as anchor point); $\tau : V \cup \{a\} \times V \rightarrow \mathbb{R}_{\geq 0}$ is a function that determines the time it takes a firefighter to move from either the depot a or a vertex $u \in V$ to any other vertex $v \in V$, and to defend v . f is the number of firefighters, and $T_S > 0$ is a constant that defines the time it takes for the fire to spread from a burnt vertex $u \in V$ to its unprotected neighbors in G .

From the assumption that the fire spreads uniformly at a constant rate, we can organize the time into time slots of length T_S . Each time slot defines a burning round, where at the beginning of each time slot, the fire propagates from burnt vertices to their neighbors in G unless they were defended by a firefighter. Once a vertex is burnt or defended, it remains in that state, and the process terminates when the fire can no longer spread. We will say that a vertex is protected if it is neither burning nor defended at the end of the process. The diffusive process starts at time slot 0 where only the vertices in F are burning.

For the case of a single firefighter, a feasible solution to the *MFP* is a valid sequence S of vertices defended by the firefighter. We say that a sequence $S = (a, u_1, \dots, u_l)$ is valid if Equation (1) holds for each $i = 1, 2, \dots, l$. In Equation (1), β_{S_i} is the time at which a vertex u_i would be burned if the defending sequence $S = (a, u_1, \dots, u_l)$ were truncated to $S_i = (a, u_1, \dots, u_{i-1})$.

The purpose of Equation (1) is to restrict the space of feasible solutions to that where the firefighter has enough time to defend every vertex in the sequence before the fire reaches them.

$$\tau(a, u_1) + \sum_{k=1}^{i-1} \tau(u_k, u_{k+1}) \leq \beta_{S_i} \tag{1}$$

Figure 1 shows examples of an instance of the *MFP* and of one of its solutions. The graph G is composed of the set of vertices $V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9\}$ and the set of edges $E = \{(v_0, v_2), (v_1, v_3), (v_2, v_3), (v_2, v_5), (v_3, v_4), (v_3, v_5), (v_3, v_6), (v_4, v_7), (v_5, v_6), (v_5, v_8), (v_6, v_9)\}$. The fire breaks out at vertices $F = \{v_0, v_1\}$, a single firefighter ($f = 1$) is initially stationed at the depot a , and $T_S = 1$. The depot and the vertices in

V are in the Euclidean plane. The number on top of the vertices indicates their shortest hop distance to their closest vertex in F , which in this example equals the smallest time slot in which they can burn. The figure also shows a feasible solution $\mathcal{S} = (a, v_5, v_6, v_7)$ in which the firefighter moves from the depot a to defend vertex v_5 , then moves to defend vertex v_6 , and lastly moves to defend vertex v_7 . For this defending sequence \mathcal{S} , the value of $\beta_{\mathcal{S}_1}$ equals 2 because vertex v_5 burns at time 2 when the truncated protecting sequence $\mathcal{S}_1 = (a)$ is applied, $\beta_{\mathcal{S}_2}$ equals 2 because vertex v_6 burns at time 2 when the truncated protecting sequence $\mathcal{S}_2 = (a, v_5)$ is applied, and similarly, $\beta_{\mathcal{S}_3}$ equals 3. \mathcal{S} is a valid sequence because $\tau(a, v_5) = 1.13 \leq \beta_{\mathcal{S}_1} = 2$, $\tau(a, v_5) + \tau(v_5, v_6) = 1.99 \leq \beta_{\mathcal{S}_2} = 2$, and $\tau(a, v_5) + \tau(v_5, v_6) + \tau(v_6, v_7) = 2.81 \leq \beta_{\mathcal{S}_3} = 3$. The process ends after three burning rounds (at time slot three) because at that time, the fire can no longer spread. The value of this solution equals five because, at the end of the process, vertices v_0, v_1, v_2, v_3 , and v_4 are burnt.

For the instance of Figure 1, the firefighter cannot choose to defend vertex v_3 because $\tau(a, v_3) = 1.74 > \beta_{\mathcal{S}'_1} = 1$ for sequence $\mathcal{S}' = (a, v_3)$. It could defend v_2 because $\tau(a, v_2) = 0.80 < \beta_{\mathcal{S}''_1} = 1$ for a sequence $\mathcal{S}'' = (a, v_2)$, but in this case no other vertex could be defended, giving a feasible solution $\mathcal{S}'' = (a, v_2)$ of value nine. In these examples, the process also ends at time slot 3 when the fire can no longer spread.

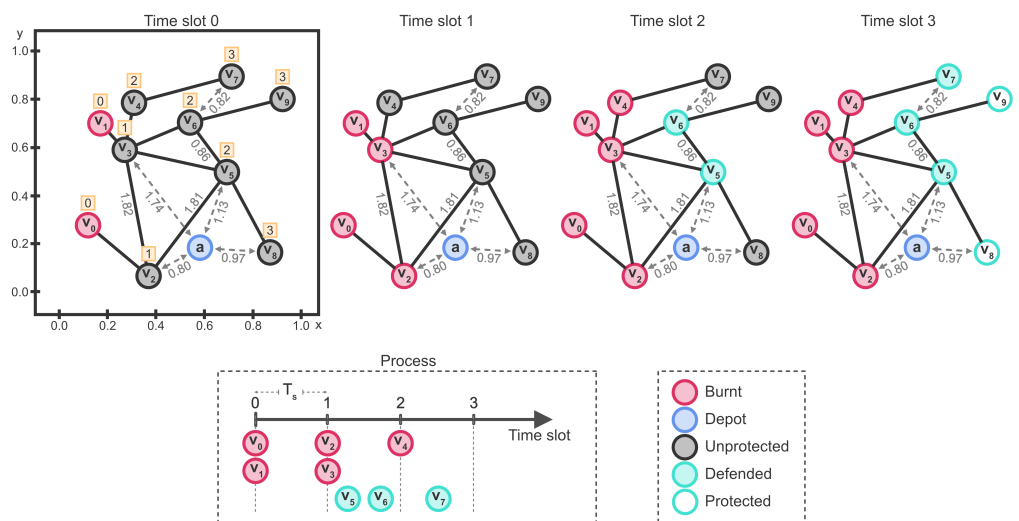


Figure 1. Example of an instance of the MFP and one of its solutions.

For the case of $f > 1$ firefighters, a feasible solution is a set of valid sequences $\mathbb{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_f\}$ of the vertices defended by each firefighter. In this case, $\beta_{\mathcal{S}_i}$ has to be extended to compute the time in which a vertex u_i in a $\mathcal{S}_x \in \mathbb{S}$ is burnt if none of the defending sequences in $\mathbb{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_f\}$ includes vertices protected at a time posterior to $\tau(a, u_1) + \sum_{k=1}^{i-1} \tau(u_k, u_{k+1})$ for $u_k \in \mathcal{S}_x$.

The objective of the minimization version of the MFP is to find a feasible solution that minimizes the number of burnt vertices at the end of the process.

The decision version of the moving firefighter problem for a single firefighter (1-MFP) is as follows.

Definition 1. Decision version of the 1-MFP problem: Given an arbitrary instance $\langle G = (V, E), F, a, \tau, 1, T_S \rangle$ of 1-MFP and an integer $k \geq 1$, is there a valid sequence \mathcal{S}_{moving} of vertices defended by the firefighter such that the number of burnt vertices at the end of the process is at most k ?

1-MFP is NP-Complete

In this section, we show that 1-MFP is NP-complete via a polynomial-time reduction (\leq_P) to the decision version of the firefighter problem for a single firefighter (1-FP), which can be defined as follows.

Definition 2. Decision version of the 1-FP problem: Given an arbitrary instance $\langle G = (V, E), F \rangle$ of 1-FP, where $F \subset V$ is the set of vertices where the fire breaks out and an integer $k \geq 1$, is there a sequence $\mathcal{S}_{original}$ of vertices defended by the firefighter such that the number of burnt vertices at the end of the process is at most k ?

We start with two preliminary lemmas.

Lemma 1. Given an instance $\langle G, F \rangle$ of 1-FP, an instance $\langle G', F', a, \tau, 1, T_S = 1 \rangle$ of 1-MFP such that $G' = G, F' = F$, defending sequences $\mathcal{S}_{original} = (u_1, \dots, u_l)$ for the instance of 1-FP, $\mathcal{S}_{moving} = (a, u'_1, \dots, u'_l)$ with $u'_i = u_i$ for each $u'_i \in \mathcal{S}_{moving}$ for the instance of 1-MFP, and function $\tau_1(u, v) = 1 \forall u \neq v$ with $u \in V \cup \{a\}, v \in V$ and $\tau_1(u, u) = 0 \forall u \in V \cup \{a\}$. The discrete time t_i in which a vertex v_i in $\mathcal{S}_{original} = (u_1, \dots, u_l)$ is defended equals the time $t'_i = i$ in which its corresponding vertex v'_i is defended in $\mathcal{S}_{moving} = (a, u'_1, \dots, u'_l)$.

Proof of Lemma 1. From the definition of τ_1 , we can observe that the time $\tau(a, u'_1) + \sum_{k=1}^{i-1} \tau(u'_k, u'_{k+1})$ in which a vertex $u'_i \in \mathcal{S}_{moving}$ is defended equals i . This is the same as the discrete-time in which the corresponding vertex u_i is defended by $\mathcal{S}_{original}$, namely, i . \square

Lemma 2. Given an instance $\langle G, F \rangle$ of 1-FP, an instance $\langle G', F', a, \tau, 1, T_S = 1 \rangle$ of 1-MFP such that $G' = G, F' = F$, defending sequences $\mathcal{S}_{original} = (u_1, \dots, u_l)$ for the instance of 1-FP, $\mathcal{S}_{moving} = (a, u'_1, \dots, u'_l)$ with $u'_i = u_i$ for each $u'_i \in \mathcal{S}_{moving}$, for the instance of 1-MFP, and function $\tau_1(u, v) = 1 \forall u \neq v$ with $u \in V \cup \{a\}, v \in V$, and $\tau_1(u, u) = 0 \forall u \in V \cup \{a\}$. A vertex $v \in G$ burns at discrete time t if and only if its corresponding vertex $v' \in G'$ burns at the beginning of time slot t .

Proof of Lemma 2. If a vertex $v \in G$ burns at discrete time t , it means that there is a shortest path P of length t , from a vertex $r \in F$ to v that does not contain any vertex in the truncated protecting sequence $\mathcal{S}_{original} = (u_1, \dots, u_t)$. Given that $G = G'$, this shortest path P also exists in G' and does not contain vertices in $\mathcal{S}_{moving} = (u'_1, \dots, u'_t)$. From Lemma 1, and because the fire propagates at the beginning of each time slot of length $T_S = 1$, vertex v' has to burn at time t , which equals the beginning of time slot t . A symmetric argument can be applied to show the other direction of the implication. \square

Theorem 1. 1-MFP is NP-complete.

Proof of Theorem 1. We first show that the 1-MFP is an NP. Given an arbitrary instance $\langle G, F, a, \tau, 1, T_S \rangle, k \geq 1$, and a defending sequence $\mathcal{S} = (a, u_1, \dots, u_l)$, a certifier algorithm can check in polynomial-time if (1) holds for each $u_i \in \mathcal{S}$, given that the firefighter defends each $u_i \in \mathcal{S}$ at time $\tau(a, u_1) + \sum_{k=1}^{i-1} \tau(u_k, u_{k+1})$. Then, the certifier algorithm can execute, also in polynomial-time, the firefighting process to verify that the number of burnt vertices is, in fact, at most k .

Now, we show that $1-FP \leq_P 1-MFP$. Given an arbitrary instance $\langle G, F \rangle$ of 1-FP and an integer k , we construct an instance of 1-MFP as $\langle G', F', a, \tau_1, 1, T_S = 1 \rangle$ with $G' = G, F' = F$, and $\tau_1(u, v) = 1 \forall u \neq v$ with $u \in V \cup \{a\}, v \in V$ and $\tau_1(u, u) = 0 \forall u \in V \cup \{a\}$. Constructing this instance of 1-MFP takes $\mathcal{O}(|V|^2)$ steps.

Next, we claim that there is a defending sequence $\mathcal{S}_{original} = (u_1, \dots, u_l)$ for the instance of the 1-FP such that at most k vertices of G burn, if and only if there is a valid defending sequence $\mathcal{S}_{moving}(a, u'_1, \dots, u'_l)$ with $u'_i = u_i$ for each $u'_i \in \mathcal{S}_{moving}$, for the instance of the 1-MFP such that at most k vertices burn in G' .

(\implies) We first show that $\mathcal{S}_{moving} = (a, u'_1, \dots, u'_l)$ is valid. Let us proceed by contradiction and assume that $u_j \in \mathcal{S}_{moving}$ is the vertex with the smallest index $j \in \{1, \dots, l\}$ such that $\tau(a, u_1) + \sum_{k=1}^{j-1} \tau(u_k, u_{k+1}) = j > \beta_{\mathcal{S}_j}$. However, from Lemma 1, we saw that the time a vertex in G' is defended by \mathcal{S}_{moving} equals the discrete time its corresponding vertex in G is defended by $\mathcal{S}_{original}$. Since vertex u_j is defended at discrete time j in the original firefighter process, it can not be burning at time j , which means that there is no path from a vertex $r \in F$ to u_j of length j or shorter that does not contain a vertex in the truncated protecting sequence (u_1, \dots, u_{j-1}) . Since $G = G'$, there is no such a path in G' , and hence $\beta_{\mathcal{S}_j} > j$, a contradiction. From Lemma 2, at the end of the burning process, exactly the same number of vertices are burning in both instances. Therefore, if at most k vertices of G burn, then at most k vertices burn in G' .

(\impliedby) Since $\mathcal{S}_{moving} = (a, u'_1, \dots, u'_l)$ is valid for the 1-MFP instance, from Lemma 1 and 2, we know that all the vertices in $\mathcal{S}_{original}$ can be defended in the 1-FP instance. From Lemma 2, we also know that if at most k vertices of G' burn, then at most k vertices of G burn. \square

Table 2 summarizes the notation used throughout the paper.

Table 2. Summary of notation.

$G = (V, E)$	Input graph.
F	Set of vertices where the fire breaks out.
a	Firefighters' depot or anchor point.
τ	Travel time function.
f	Number of firefighters.
T_S	Length of the time slots (burning round).
$\langle G, F, a, \tau, f, T_S \rangle$	Instance of the 1-MFP.
$\langle G, F \rangle$	Instance of the 1-FP.
$\mathcal{S} = (a, u_1, \dots, u_l)$	Defending sequence.
$\beta_{\mathcal{S}_i}$	Time when vertex u_i burns, given the truncated defending sequence $\mathcal{S}_i = (a, u_1, \dots, u_{i-1})$.
$\mathbb{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_f\}$	Set of defending sequences for $f > 1$.
B	Upper bound on the number of burning rounds (or time slots).
D	Upper bound on the number of vertices defended within a single burning round.
$b_{i,j}$	Binary variable that indicates whether v_i is burning at the j -th burning round.
$d_{i,j}$	Binary variable that indicates whether v_i is defended at the j -th burning round.
$d'_{j,i,k}$	Binary variable that indicates whether v_i is defended at the k -th defense round of the j -th burning round.
$p_{j,i,k}$	Binary variable that indicates whether v_i was the last defended vertex at the k -th defense round of the j -th burning round.
t_j	Continuous variable that indicates the time needed to carry out a defending sequence within the j -th burning round.
\mathbf{b}	$n \times B$ matrix that contains the $b_{i,j}$ binary variables.
\mathbf{d}	$n \times B$ matrix that contains the $d_{i,j}$ binary variables.
\mathbf{d}'_j	$n \times D$ matrix that contains the $d'_{j,i,k}$ binary variables.
\mathbf{p}_j	$n \times D$ matrix that contains the $p_{j,i,k}$ binary variables.

4. Mixed-Integer Quadratically Constrained Program for the 1-MFP

Expressions (2) to (23) define a mixed-integer quadratically constrained program (MIQCP) for the optimization version of the 1-MFP. This MIQCP is based on the integer linear program (ILP) for the original firefighter problem (FP). For simplicity, let us assume that vertices are labeled as $\{v_1, v_2, \dots, v_n\}$. In addition, let us consider the anchor point, a , as an isolated vertex in the graph. This way, the fire cannot catch this special vertex, and inconsistencies are not added to the formulation. As in the original FP, the proposed MIQCP for the 1-MFP requires an upper bound B on the number of burning rounds (or time slots) at which the diffusive process finishes. However, unlike the original FP, it also requires an upper bound D on the number of vertices that are defended within a single burning round. We will say that defending a vertex during a burning round defines a defense round within that burning round. The length of the defense rounds is a function of τ , and it is possible to have from 0 to D defense rounds in a given burning round.

The ILP for the original FP is relatively simple; it has $\mathcal{O}(nB)$ binary variables and $\mathcal{O}(nB\Delta)$ constraints, where Δ is the graph’s largest vertex degree. In contrast, our proposed MIQCP for the 1-MFP has $\mathcal{O}(nBD)$ binary variables, $\mathcal{O}(\max\{|E|D, nBD\})$ linear constraints, and $\mathcal{O}(nBD)$ quadratic constraints. Since B and D are upper bounded by n , the numbers of variables and constraints are upper bounded by $\mathcal{O}(n^3)$. In addition, this MIQCP assumes that function τ is normalized with respect to the length of the burning rounds (time slots), and hence that $T_S = 1$.

$$\min \sum_{i=1}^n b_{i,B} \tag{2}$$

$$\text{s.t. } b_{i,j} \geq b_{i,j-1} \quad \forall v_i \in V, \forall j \in [1, B] \tag{3}$$

$$d_{i,j} \geq d_{i,j-1} \quad \forall v_i \in V, \forall j \in [1, B] \tag{4}$$

$$b_{i,j} + d_{i,j} \leq 1 \quad \forall v_i \in V, \forall j \in [1, B] \tag{5}$$

$$b_{i,j} + d_{i,j} \geq b_{k,j-1} \quad \forall v_i \in V, \forall j \in [1, B], \forall v_k \in N(v_i) \tag{6}$$

$$d'_{j,i,1} \geq d_{i,j-1} \quad \forall v_i \in V, \forall j \in [1, B] \tag{7}$$

$$d'_{j,i,D} = d_{i,j} \quad \forall v_i \in V, \forall j \in [1, B] \tag{8}$$

$$d'_{j,i,k} \geq d'_{j,i,k-1} \quad \forall v_i \in V, \forall j \in [1, B], \forall k \in [2, D] \tag{9}$$

$$p_{j,i,k} \geq d'_{j,i,k} - d'_{j,i,k-1} \quad \forall v_i \in V, \forall j \in [1, B], \forall k \in [2, D] \tag{10}$$

$$p_{j,i,1} \geq d'_{j,i,1} - d_{i,j-1} \quad \forall v_i \in V, \forall j \in [1, B] \tag{11}$$

$$\sum_{i=1}^n p_{j,i,k} = 1 \quad \forall k \in [1, D], \forall j \in [1, B] \tag{12}$$

$$p_{j,i,k} \geq p_{j,i,k-1} \cdot \left(1 - \sum_{l=1}^n (d'_{j,l,k} - d'_{j,l,k-1}) \right) \quad \forall v_i \in V, \forall j \in [1, B], \forall k \in [2, D] \tag{13}$$

$$p_{j,i,1} \geq p_{j-1,i,D} \cdot \left(1 - \sum_{l=1}^n (d'_{j,l,1} - d_{l,j-1}) \right) \quad \forall v_i \in V, \forall j \in [1, B] \tag{14}$$

$$t_j = t_{j-1} + \sum_{l=1}^n \left(\sum_{i=1}^n p_{j,i,1} \cdot \tau_{l,i} \right) \cdot p_{j-1,l,D} \quad \forall j \in [1, B] \tag{15}$$

$$+ \sum_{k=2}^D \left(\sum_{l=1}^n \left(\sum_{i=1}^n p_{j,i,k} \cdot \tau_{l,i} \right) \cdot p_{j,l,k-1} \right) \tag{16}$$

$$t_j \leq j \quad \forall j \in [1, B] \tag{16}$$

$$\text{where } t_0 = 0 \tag{17}$$

$$d_{a,0} = 1 \tag{18}$$

$$d_{i,0} = 0 \quad \forall v_i \in V \setminus \{a\} \tag{19}$$

$$b_{i,0} = \begin{cases} 1 & \text{if } i \in F \\ 0 & \text{if } i \notin F \end{cases} \quad \forall v_i \in V \tag{20}$$

$$d'_{0,i,k} = p_{0,i,k} = d_{i,0} \quad \forall v_i \in V, \forall k \in [1, D] \tag{21}$$

$$b_{i,j}, d_{i,j}, p_{j,i,k}, d'_{j,i,k} \in \{0, 1\} \quad \forall v_i \in V, \forall j \in [1, B], \forall k \in [1, D] \tag{22}$$

$$t_j \in \mathbb{R}^+ \quad \forall j \in [1, B] \tag{23}$$

The ILP for the FP problem only has two sets of binary variables, $b_{i,j}$ and $d_{i,j}$. Similarly, the main variables of the proposed MIQCP for the 1-MFP are $b_{i,j}$ and $d_{i,j}$ too. These variables report the state of each vertex i at the j th burning round, where $i \in [1, n]$ and $j \in [1, B]$. Thus, if a vertex v_i is burnt at round j , then $b_{i,j} = 1$; otherwise, $b_{i,j} = 0$. If vertex v_i is burnt

at round j , it is because of one of the following reasons: it was originally burnt at some previous round (Constraint (3)), or it is in the neighborhood of some previously burnt vertex (Constraint (6)). In case vertex v_i is defended at round j , then $d_{i,j} = 1$; otherwise, $d_{i,j} = 0$. A vertex that is defended at any round will remain in that state (Constraint (4)), and if a vertex is in the neighborhood of some previously burnt vertex, then it has to be either burned or defended (Constraint (6)). By Constraint (5), a vertex cannot be simultaneously burnt and defended. For further clarification, we can organize these variables in two $n \times B$ matrices (see Equations (24) and (25)), where matrix \mathbf{d} can be further extended to matrices \mathbf{d}'_j and \mathbf{p}_j (see Equations (26) and (27)), where $j \in [1, B]$.

$$\mathbf{b} = \begin{pmatrix} b_{1,0} & b_{1,1} & \cdots & b_{1,B} \\ b_{2,0} & b_{2,1} & \cdots & b_{2,B} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n,0} & b_{n,1} & \cdots & b_{n,B} \end{pmatrix} \tag{24}$$

$$\mathbf{d} = \begin{pmatrix} d_{1,0} & d_{1,1} & \cdots & d_{1,B} \\ d_{2,0} & d_{2,1} & \cdots & d_{2,B} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n,0} & d_{n,1} & \cdots & d_{n,B} \end{pmatrix} \tag{25}$$

Given that the 1-MFP allows sequentially defending many vertices within a burning round, we added a set of variables $d'_{j,i,k}$ in order to report the sequence of defended vertices, where $j \in [1, B]$, $i \in [1, n]$, and $k \in [1, D]$. In more detail, if vertex v_i is defended at the k th defense round, within the j th burning round, then $d'_{j,i,k} = 1$; otherwise, $d'_{j,i,k} = 0$. Notice that D is an upper bound on the number of defended vertices within one burning round. We can organize these variables into B matrices of size $n \times D$. Thus, there is a matrix \mathbf{d}'_j for each j th burning round (see Equation (26)).

$$\mathbf{d}'_j = \begin{pmatrix} d'_{1,1} & d'_{1,2} & \cdots & d'_{1,D} \\ d'_{2,1} & d'_{2,2} & \cdots & d'_{2,D} \\ \vdots & \vdots & \ddots & \vdots \\ d'_{n,1} & d'_{n,2} & \cdots & d'_{n,D} \end{pmatrix} \tag{26}$$

By Constraint (7), all vertices defended at the $(j - 1)$ th burning round stay defended at the beginning of the first defense round of the following burning round, which is reported in the first column of \mathbf{d}'_j . By Constraint (8), the state of the defended vertices at the end of the j th burning round (column D of \mathbf{d}'_j) is copied into column j of matrix \mathbf{d} . In other words, each column j of matrix \mathbf{d} is a summary of what happens inside each matrix \mathbf{d}'_j . Similarly to Constraint (4), Constraint (9) indicates that once a vertex is defended at the k th defense round of the j th burning round, it will stay defended.

Constraints (10)–(12) describe the relationship between the binary variables $p_{j,i,k}$ and variables $d'_{j,i,k}$. The goal of \mathbf{p}_j is to report the last defended vertex at each defense round within each burning round. Notice that $p_{j,i,k} = 1$ might mean two different things: vertex v_i has been defended precisely at the k th defense round within burning round j (Constraints (10) and (11)), or vertex v_i is the last vertex defended at some previous round. Constraint (12) guarantees that exactly one vertex is the last defended at each defense round. Please note that Constraint (18) establishes that the anchor point a is defended at the beginning of the process. For convenience, we can organize these variables in a matrix \mathbf{p}_j for each burning round (see Equation (27)).

Constraints (13) and (14) guarantee that the information about the last defended vertex is copied into the following columns of \mathbf{p} . When a new vertex becomes defended, these constraints are deactivated by making the right-hand side equal to zero. In this case, the

subsequent columns are no longer a copy of the previous ones; instead, the currently defended vertex is updated. Constraints (15) and (16) guarantee that the time t_j needed to carry out a sequence of defended vertices within the j th burning round does not exceed the current number j of time slots of the diffusive process. By these two constraints, Constraints (3) and (5)–(14), which define the relationship between the defended vertices and the dynamics of the diffusive process, the firefighter cannot defend a vertex that the fire has already spread to (which is consistent with Equation (1)).

$$P_j = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,D} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,D} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n,1} & p_{n,2} & \cdots & p_{n,D} \end{pmatrix} \tag{27}$$

The solution to an 1-MFP instance is obtained from the d' matrices by sequentially scanning them for the first time that a value of one appears in a row of one of such matrices. Remember that the first time that a value of one appears in row i of the matrix d'_j indicates that the vertex v_i is defended at burning round j , and hence, that it is part of the resulting defending sequence \mathcal{S} . The ordering of the vertices in \mathcal{S} is obtained from the burning and defense rounds where they were defended.

For further clarification, Figure 2 shows the value of each variable in the solution of the example instance from Figure 1. In this example, we follow the convention of placing the anchor point in the last row of the matrices. Column $j = 0$ represents the process' initial conditions where vertices v_0 and v_1 are the initial fire sources and the firefighter is located at the anchor point. The remaining columns of matrices b and d report the state of the vertices at the subsequent burning rounds. The resulting defending sequence \mathcal{S} equals (a, v_5, v_6, v_7) because the first value of one appears in the sixth row (the one corresponding to vertex v_5) of the first column of d'_2 , the second value of one appears in the seventh row of d'_3 , and the last value of one appears in the eighth row of the first column of d'_3 . Time $t_1 = 0$ because no vertex is defended during the first burning round while $t_2 = 1.13 + 0.86 = 1.99$ because the firefighter moves from the anchor a to defend vertex v_5 and then from v_5 to defend v_6 , which takes 1.99 units of time. Similarly, $t_3 = 2.81$ because that is the time needed to complete the whole defending sequence $\mathcal{S} = (a, v_5, v_6, v_7)$, which is optimal for this instance.

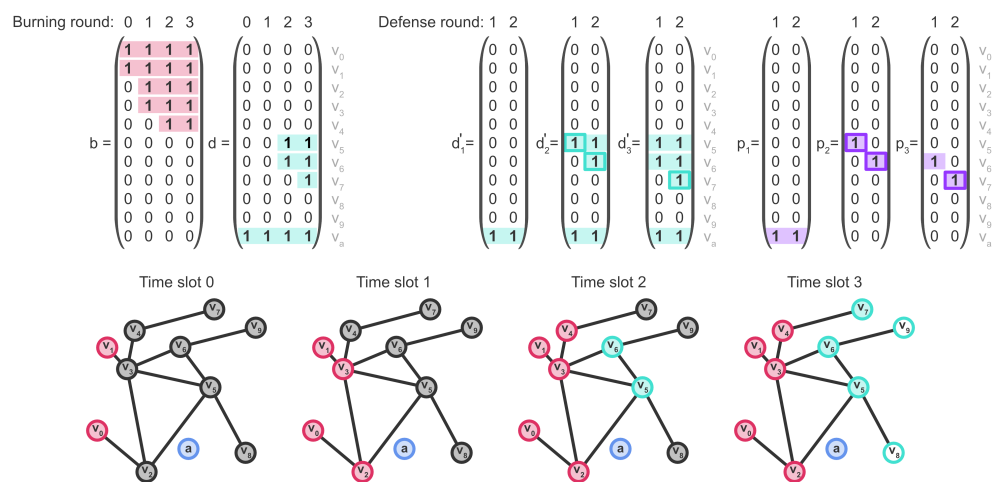


Figure 2. State of the matrices defined by the MIQCP for the optimal solution of the instance defined in Figure 1. Here, $T_S = 1$, $B = 3$, and $D = 2$.

Regarding a MIQCP for the MFP with more than one firefighter, we conjecture that the number of variables and constraints might increase by an order of f . The precise formulation is left as future work.

5. Experimental Performance Evaluation

In this section, we present the results of a series of experiments that characterize the performance of the proposed MIQCP for the 1-MFP. The purpose was twofold: first, to evaluate the impacts of the size of the graph, the scale of function τ , and the number of vertices where the fire breaks out on the value of the objective function and the MIQCP's running time; second, to provide a set of optimally solved instances that can be used for future performance comparison analyses of heuristic or approximated solutions.

For these experiments, we generated instances $\langle G, F, a, \tau, f, T_S \rangle$ of the 1-MFP, where the number of firefighters f equals one, the length of the time slots T_S also equals one, the number of initial fires $|F|$ is in $\{1, 3\}$, the graph $G = (V, E)$ with $|V| = n \in \{10, 20, 30, 40\}$ is randomly generated, and its vertices and the anchor point are placed uniformly at random inside a $1 \times 1 \times 1$ cube. To obtain a random graph G of size n , we generated a series of graphs $G' = (V', E')$ of size $n \leq |V'| < n + \lceil n/4 \rceil$ following the Erdős–Rényi model with $p = 2.5/n$, until finding a G' with a connected component of size n which is used as G . Lastly, τ is obtained from the Euclidean distances between the locations of the vertices and the anchor point.

We implemented the proposed MIQCP using Python and a commercial version of Gurobi 9.5.1 [38]. Although there are other optimization solvers, not all of them can solve MIQCP problems. Gurobi and CPLEX [39] are among the best solvers with such capabilities. From them, we selected Gurobi because it is one of the fastest optimization solvers to date. All the experiments were executed using a computer running Ubuntu 20.04 on an Intel Core i7-6700 processor of 3.40 GHz with 32 GB of RAM.

For the results presented here, we used optimal values of the two hyperparameters introduced in the MIQCP—namely, the upper bound B for the number of time slots needed to complete the diffusive process and the upper bound D for the maximum number of vertices that can be defended in a single time slot. These optimal values were determined experimentally for each instance. As discussed in Section 4, the values of these upper bounds significantly impact the performance of the MIQCP because they determine the number of constraints. It is also important to highlight that underestimating these values precludes the MIQCP from finding optimal and even correct solutions.

Figure 3 presents the running times needed to solve instances with increasing numbers of vertices and initial fires, and when scaling function τ by a factor $\lambda \in \{2, 7\}$. The solid lines represent the average value computed over ten different random instances, and the shaded region represents the confidence interval for a confidence level of 95%. The results are as expected. From the figure, we can observe that the running time increases rapidly as the size of the graphs (n) increases because the numbers of linear ($\mathcal{O}(\max\{|E|D, nBD\})$) and quadratic ($\mathcal{O}(nBD)$) constraints increase as a function of n . We can also observe that the running time tends to be smaller with a larger number of initial burnt vertices ($|F|$) because the upper bound B on the number of time slots needed to complete the diffusive process also tends to be smaller as $|F|$ increases. The latter is because B is proportional to the length of the shortest paths from vertices in F to the remaining vertices in G . Lastly, the running time decreases as the value of λ increases because the upper bound D on the number of vertices that can be defended in a single time slot decreases.

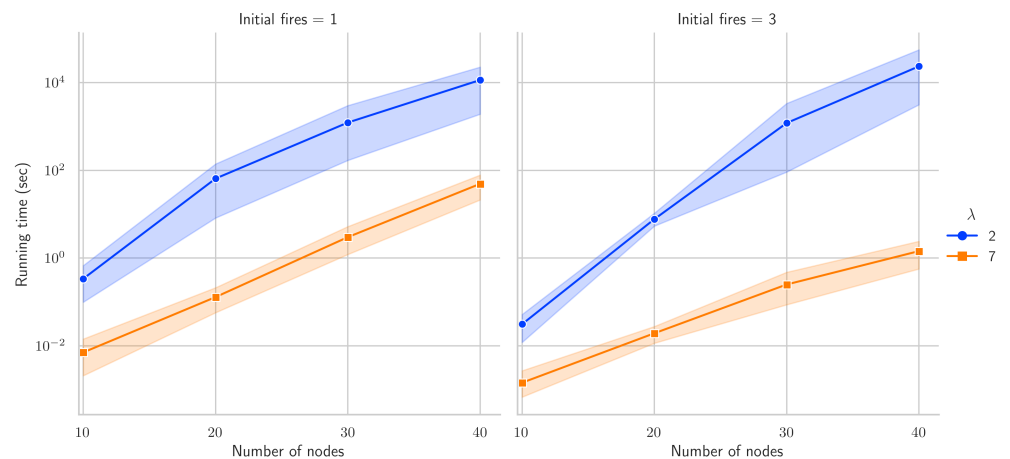


Figure 3. Running time with one initial fire (left) and three initial fires (right). The x axis shows the number of vertices in the input graph, and the y axis shows the running time in seconds. The solid lines represent the average values over ten random instances, and the shaded regions are the confidence intervals for a confidence level of 95%.

Figure 4 shows the average number of burnt vertices at the end of the diffusive processes when increasing the number of vertices, the number of initial fires, and the value of λ . As expected, the number of burnt vertices increases as the size of the graphs increases because, in general, the fire can propagate to a larger number of vertices. Increasing the number of initial fires increases the number of burnt vertices because the firefighter has to contain more than one diffusive process. This also reduces the objective function’s variability because having more than one initial fire makes the diffusive process less sensitive to the topological properties of the neighborhood where the fire breaks out. Lastly, the number of burnt vertices increases as the value of λ increases because this has the effect of making the firefighter slower.

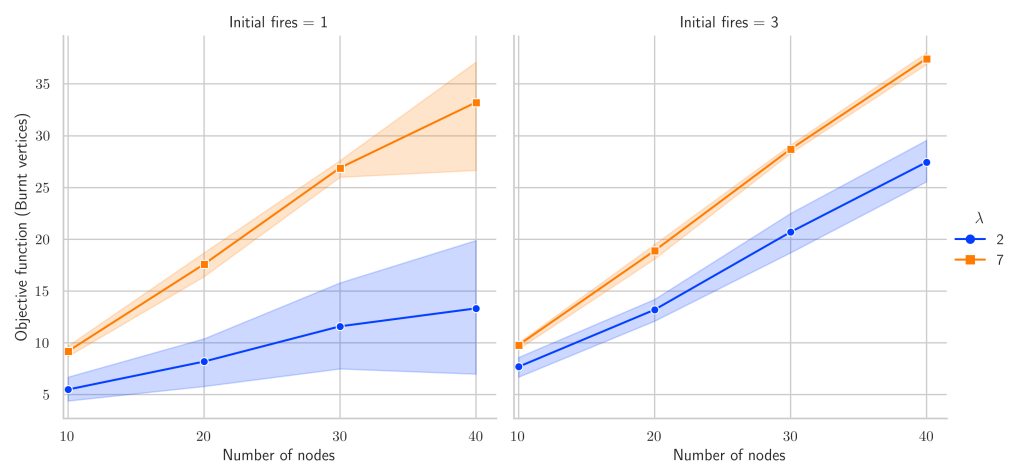


Figure 4. Numbers of burnt vertices with one initial fire (left) and three initial fires (right). The x axis shows the number of vertices in the input graph, and the y axis shows the number of burnt vertices at the end of the process. The solid lines represent the average values over ten random instances, and the shaded regions are the confidence intervals for a confidence level of 95%.

We performed experiments for graphs of size $n = 50$, but unfortunately, most of the executions did not end within a predefined time budget of 24 h, and in many cases, experienced memory crashes. This indicates a need for more scalable solutions for the MFP.

6. Conclusions

We presented the moving firefighter problem (MFP), a generalization of the original firefighter problem (FP) presented by Hartnell, which incorporates a function τ that determines the time it takes a firefighter to move from vertex u to vertex v in the input graph to defend v . We argue that this generalization allows modeling more realistic situations where the time needed to defend an entity modeled by a vertex is not the same for all the entities and where there is either a physical or logical distance between each pair of entities. This new formulation has practical applications for studying the spread and containment dynamics in networks.

Using polynomial time reduction from the decision version of the original firefighter problem for a single firefighter 1-FP, we showed that the decision version of the moving firefighter problem for a single firefighter 1-MFP is indeed a generalization of the 1-FP, and hence, that it belongs to the NP-complete class of problems.

We presented a mixed-integer quadratically constrained program (MIQCP) for the optimization version of the 1-MFP with metric functions τ that have $\mathcal{O}(nBD)$ binary variables, $\mathcal{O}(B)$ real-valued variables, $\mathcal{O}(\max\{|E|D, nBD\})$ linear constraints, and $\mathcal{O}(nBD)$ quadratic constraints. B and D are upper bounds to the numbers of burning and defense rounds, respectively. To empirically characterize the performance of the MIQCP, we solved 160 randomly generated instances. The results revealed that the running time of the MIQCP strongly depends on finding adequate values for B and D .

Future work includes designing scalable solutions for the 1-MFP, solutions for instances with more than one firefighter, and solutions specifically tailored for particular topologies such as trees. We also intend to design approximation algorithms and characterize the approximability of the MFP.

Author Contributions: Conceptualization, B.R.G.-D.-L.-P., J.G.-D., R.M.-M. (Rolando Menchaca-Méndez), M.A.M.-M., R.M.-M. (Ricardo Menchaca-Méndez) and O.A.G.-D.-L.-P.; Methodology, B.R.G.-D.-L.-P., J.G.-D., R.M.-M. (Rolando Menchaca-Méndez), M.A.M.-M., R.M.-M. (Ricardo Menchaca-Méndez) and O.A.G.-D.-L.-P.; Software, B.R.G.-D.-L.-P., J.G.-D., M.A.M.-M. and O.A.G.-D.-L.-P.; Validation, R.M.-M. (Ricardo Menchaca-Méndez); Formal analysis, B.R.G.-D.-L.-P., J.G.-D., R.M.-M. (Rolando Menchaca-Méndez), M.A.M.-M., R.M.-M. (Ricardo Menchaca-Méndez) and O.A.G.-D.-L.-P.; Writing—original draft, B.R.G.-D.-L.-P., J.G.-D., R.M.-M. (Rolando Menchaca-Méndez), M.A.M.-M., R.M.-M. (Ricardo Menchaca-Méndez) and O.A.G.-D.-L.-P.; Writing—review & editing, B.R.G.-D.-L.-P., J.G.-D., R.M.-M. (Rolando Menchaca-Méndez), M.A.M.-M., R.M.-M. (Ricardo Menchaca-Méndez) and O.A.G.-D.-L.-P.; Visualization, B.R.G.-D.-L.-P., J.G.-D. and O.A.G.-D.-L.-P.; Supervision, R.M.-M. (Rolando Menchaca-Méndez) and R.M.-M. (Ricardo Menchaca-Méndez); Project administration, R.M.-M. (Rolando Menchaca-Méndez); All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the “Consejo Nacional de Ciencia y Tecnología” of México and the “Instituto Politécnico Nacional.”

Data Availability Statement: The source code used in Section 5 is available at <https://github.com/BrunoGupa/MovingFirefighterProblem>. The same repository contains the specific instances and their corresponding optimal solutions that were used to obtain the results presented in Section 5.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Hartnell, B. Firefighter! an application of domination. In Proceedings of the the 24th Manitoba Conference on Combinatorial Mathematics and Computing, University of Manitoba, Winnipeg, ON, Canada, 30 September–2 October 1994.
2. Finbow, S.; MacGillivray, G. The Firefighter Problem: a survey of results, directions and questions. *Australas. J. Comb.* **2009**, *43*, 57–78.
3. Khakzad, N. A graph theoretic approach to optimal firefighting in oil terminals. *Energies* **2018**, *11*, 3101. [[CrossRef](#)]
4. Wang, K.; Yuan, Y.; Chen, M.; Lou, Z.; Zhu, Z.; Li, R. A Study of Fire Drone Extinguishing System in High-Rise Buildings. *Fire* **2022**, *5*, 75. [[CrossRef](#)]
5. Fogarty, P. Catching the Fire on Grids. Master’s Thesis, Department of Mathematics, University of Vermont, Burlington, VT, USA, 2003.

6. Develin, M.; Hartke, S.G. Fire containment in grids of dimension three and higher. *Discret. Appl. Math.* **2007**, *155*, 2257–2268. [[CrossRef](#)]
7. Ng, K.L.; Raff, P. A generalization of the firefighter problem on $Z \times Z$. *Discret. Appl. Math.* **2008**, *156*, 730–745. [[CrossRef](#)]
8. Anshelevich, E.; Chakrabarty, D.; Hate, A.; Swamy, C. Approximation algorithms for the firefighter problem: Cuts over time and submodularity. In *Proceedings of the International Symposium on Algorithms and Computation*; Springer: Berlin, Germany, 2009; pp. 974–983.
9. King, A.; MacGillivray, G. The firefighter problem for cubic graphs. *Discret. Math.* **2010**, *310*, 614–621. [[CrossRef](#)]
10. Bonato, A.; Messinger, M.E.; Prałat, P. Fighting constrained fires in graphs. *Theor. Comput. Sci.* **2012**, *434*, 11–22. [[CrossRef](#)]
11. Bazgan, C.; Chopin, M.; Ries, B. The firefighter problem with more than one firefighter on trees. *Discret. Appl. Math.* **2013**, *161*, 899–908. [[CrossRef](#)]
12. Klein, R.; Levkopoulos, C.; Lingas, A. Approximation algorithms for the geometric firefighter and budget fence problems. In *Proceedings of the Latin American Symposium on Theoretical Informatics*; Springer: Berlin, Germany, 2014; pp. 261–272.
13. Anshelevich, E.; Chakrabarty, D.; Hate, A.; Swamy, C. Approximability of the firefighter problem. *Algorithmica* **2012**, *62*, 520–536. [[CrossRef](#)]
14. Michalak, K.; Knowles, J.D. Simheuristics for the multiobjective nondeterministic firefighter problem in a time-constrained setting. In *Proceedings of the European Conference on the Applications of Evolutionary Computation*; Springer: Berlin, Germany, 2016; pp. 248–265.
15. Klein, R.; Levkopoulos, C.; Lingas, A. Approximation algorithms for the geometric firefighter and budget fence problems. *Algorithms* **2018**, *11*, 45. [[CrossRef](#)]
16. Coupechoux, P.; Demange, M.; Ellison, D.; Jouve, B. Firefighting on trees. *Theor. Comput. Sci.* **2019**, *794*, 69–84. [[CrossRef](#)]
17. Zambon, M.J.; de Rezende, P.J.; de Souza, C.C. Solving the geometric firefighter routing problem via integer programming. *Eur. J. Oper. Res.* **2019**, *274*, 1090–1101. [[CrossRef](#)]
18. Farhadi, M.; Toriello, A.; Tetali, P. The Traveling Firefighter Problem. In *Proceedings of the SIAM Conference on Applied and Computational Discrete Algorithms (ACDA21)*, Philadelphia, PA, USA, 19–21 July 2021; pp. 205–216.
19. Granwal, L. Area of Bushfire Damage in Australia 2020 by Land Type. 2022. Available online: <https://www.statista.com/statistics/1093799/australia-bushfire-damage-area-by-land-type/> (accessed on 14 april 2022).
20. Sönnichsen, N. Acres Burned by Wildfires in the United States from 1983 to 2020. 2021. Available online: <https://www.statista.com/statistics/203990/area-of-acres-burnt-due-to-wildland-fires-in-the-us/> (accessed on 14 april 2022).
21. Alves, B. Number of Wildfires in Brazilian Biomes in 2021, by Type of Biome. 2022. Available online: <https://www.statista.com/statistics/1044209/number-wildfires-brazil-biome/> (accessed on 14 april 2022).
22. MacGillivray, G.; Wang, P. On the firefighter problem. *JCMCC J. Comb. Math. Comb. Comput.* **2003**, *47*, 57–78.
23. Chlebíková, J.; Chopin, M. The firefighter problem: Further steps in understanding its complexity. *Theor. Comput. Sci.* **2017**, *676*, 42–51. [[CrossRef](#)]
24. Fomin, F.V.; Heggenes, P.; van Leeuwen, E.J. The firefighter problem on graph classes. *Theor. Comput. Sci.* **2016**, *613*, 38–50. [[CrossRef](#)]
25. Finbow, S.; King, A.; MacGillivray, G.; Rizzi, R. The firefighter problem for graphs of maximum degree three. *Discret. Math.* **2007**, *307*, 2094–2105. [[CrossRef](#)]
26. Hartnell, B.; Li, Q. Firefighting on Trees: How Bad is the Greedy Algorithm? In *Proceedings of the Thirty-First Southeastern International Conference on Combinatorics, Graph Theory and Computing*, Congressus Numerantium 145, Boca Raton, FL, USA, 11 September 2000; pp. 187–192.
27. Cai, L.; Verbin, E.; Yang, L. Firefighting on trees: $(1 - 1/e)$ -approximation, fixed parameter tractability and a subexponential algorithm. In *Proceedings of the International Symposium on Algorithms and Computation*; Springer: Berlin, Germany, 2008; pp. 258–269.
28. Iwaikawa, Y.; Kamiyama, N.; Matsui, T. Improved approximation algorithms for firefighter problem on trees. *IEICE Trans. Inf. Syst.* **2011**, *94*, 196–199. [[CrossRef](#)]
29. Adjashvili, D.; Baggio, A.; Zenklusen, R. Firefighting on trees beyond integrality gaps. *ACM Trans. Algorithms (TALG)* **2018**, *15*, 1–33.
30. Bazgan, C.; Chopin, M.; Cygan, M.; Fellows, M.R.; Fomin, F.V.; Van Leeuwen, E.J. Parameterized complexity of firefighting. *J. Comput. Syst. Sci.* **2014**, *80*, 1285–1297. [[CrossRef](#)]
31. Bonato, A.; Janssen, J.; Roshanbin, E. Burning a graph as a model of social contagion. In *Proceedings of the International Workshop on Algorithms and Models for the Web-Graph*; Springer: Berlin, Germany, 2014; pp. 13–22.
32. Bessy, S.; Bonato, A.; Janssen, J.; Rautenbach, D.; Roshanbin, E. Burning a graph is hard. *Discret. Appl. Math.* **2017**, *232*, 73–87. [[CrossRef](#)]
33. García-Díaz, J.; Pérez-Sansalvador, J.C.; Rodríguez-Henríquez, L.M.X.; Cornejo-Acosta, J.A. Burning graphs through farthest-first traversal. *IEEE Access* **2022**, *10*, 30395–30404. [[CrossRef](#)]
34. Kempe, D.; Kleinberg, J.; Tardos, É. Maximizing the spread of influence through a social network. In *Proceedings of the Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, USA, 24–27 August 2003; pp. 137–146.

35. Banerjee, S.; Jenamani, M.; Pratihar, D.K. A survey on influence maximization in a social network. *Knowl. Inf. Syst.* **2020**, *62*, 3417–3455. [[CrossRef](#)]
36. López-García, Á.O.; Rodríguez-Gómez, G.; López-López, A. A Physarum-Inspired Approach for Influence Maximization. In *Proceedings of the International Conference on Metaheuristics and Nature Inspired Computing*; Springer: Berlin, Germany, 2021; pp. 18–32.
37. García-Díaz, J.; Rodríguez-Henríquez, L.M.X.; Pérez-Sansalvador, J.C.; Pomares-Hernández, S.E. Graph Burning: Mathematical Formulations and Optimal Solutions. *Mathematics* **2022**, *10*, 2777. [[CrossRef](#)]
38. QMKPy: A Python Testbed for the Quadratic Multiple Knapsack Problem. *arXiv* **2022**, arXiv:2211.17222.
39. A two-stage algorithm for the Virtual Network Embedding problem. In *Proceedings of the 2021 IEEE 46th Conference on Local Computer Networks (LCN)*, Edmonton, AB, Canada, 4–7 October 2021.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.