Computers & Industrial Engineering 168 (2022) 108139

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Improved solution of the Budget constrained Fuel Treatment Scheduling problem and extensions

Federico Della Croce [a,b], Marco Ghirardi [a], Rosario Scatamacchia [a,*]

[a] Dipartimento di Ingegneria Gestionale e della Produzione, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy
[b] CNR, IEIIT, Torino, Italy

ARTICLE INFO

Keywords:
OR in natural resources
Fuel management
Prescribed burning
Budget constrained fuel treatment scheduling
Matheuristics

ABSTRACT

This paper considers the budget constrained fuel treatment scheduling (BFTS) problem where, in the context of wildfire mitigation, the goal is to inhibit the potential of fire spread in a landscape by proper fuel treatment activities. Given a time horizon represented by consecutive unit periods, the landscape is divided into cells and represented as a grid graph where each cell has a fuel age that increases over time and becomes old if no treatment is applied in the meantime: this induces a potential high fire risk whenever two contiguous cells are old. Cells fuel ages can be reset to zero under appropriate fuel treatments but there is a limited budget for treatment in each period. The problem calls for finding a suitable selection of cells to be treated so as to minimize the presence of old contiguous cells over the whole time horizon. We prove that problem BFTS is strongly *NP*-complete on paths and thus on grid graphs and show that no polynomial time approximation algorithm exists unless $\mathcal{P} = \mathcal{NP}$. We provide an enhanced integer linear programming formulation of the problem with respect to the relevant literature that shows up to be efficiently solved by an ILP solver on reasonably large size instances. Finally, we consider a harder periodic variant of the problem with the aim of finding a cyclic treatment plan with cycles of length T and propose a matheuristic approach capable of efficiently tackling those instances where an ILP solver applied to the ILP formulation runs into difficulties.

1. Introduction

One of the most studied problems in wildfire mitigation is Fuel Management. Fuel Management aims to reduce potential fire intensity and fire spread in vegetation areas. Recent interest in fire and fuel management is particularly motivated by short fire return intervals and new zones where fire was excluded during the 20th century. This creates a need for the long-term reduction of fuel loads. The particularity of this management problem is that it requires new and spatially explicit management science methods. The objective of Fuel Management is the modification of potential fire behavior or fire effects by undertaking a minimum action on its fuel. The main idea consists in applying a treatment on selected specific areas, either by harvesting or burning according to some ranking of risk or effectiveness. In particular, several studies recognized the effectiveness of prescribed burning in reducing fire intensity and severity of wildfires (see, e.g., Boer, Sadler, Wittkuhn, McCaw, & Grierson, 2009; Fernandes & Botelho, 2003). As stated in Stephens and Ruth (2005), since the end of the last century the US Forest Service has slowly incorporated burning practices into its forest management policies. Similarly, there are several examples of prescribed burning activities in Australia. As stated in Mackenzie et al. (2021), the NSW National Parks and Wildlife Service conducts regular prescribed burning for bushfire hazard reduction within Wollemi National Park. Besides, this work was originated by the EC Marie Curie project GEOSAFE (https://geosafe.lessonsonfire.eu/) for wildfires management and containment in Europe and in Australia. Several fuel management optimization problems and related solution techniques were considered in recent years. We mention among others one of the first papers in literature applying mixed integer linear programming (MILP) in the context of wildfire fuel management (Wei, Rideout, & Kirsch, 2008). Since then, many operations research approaches have been proposed to schedule fuel treatment activities (see, e.g., Minas, Hearne, & Handmer, 2012). Here we focus on a strongly investigated problem in the field denoted as Budget constrained Fuel Treatment Scheduling (BFTS). This problem was firstly tackled by means of MILP modeling in Minas, Hearne, and Martell (2014) and gave rise to several publications on related generalized models in Leon, Reijnders, Hearne, Ozlen, and Reinke (2019), Rachmawati, Ozlen, Hearne, and Reinke

* Corresponding author.
E-mail addresses: federico.dellacroce@polito.it (F. Della Croce), marco.ghirardi@polito.it (M. Ghirardi), rosario.scatamacchia@polito.it (R. Scatamacchia).

https://doi.org/10.1016/j.cie.2022.108139
Received 24 November 2020; Received in revised form 25 February 2022; Accepted 28 March 2022
Available online 1 April 2022
0360-8352/

(2018) and Rachmawati, Ozlen, Reinke, and Hearne (2016). We refer to the mentioned works for a discussion on the practical applications of the models. Some complexity and approximation findings on fuel treatment scheduling on graphs were discussed in Demange and Tanasescu (2015). The BFTS problem is a multi-period problem where the goal is to inhibit the potential of fire spread in a landscape by proper fuel treatment activities (such as prescribed burning). The landscape can be divided in cells for fuel treatments according to some features of interest (e.g. type of vegetation) or operational conditions. Each cell has a fuel age that increases over time. A cell becomes "old" if its fuel age gets larger than the inhibition period caused by a treatment. High fire risk may occur in the landscape whenever two contiguous vegetation areas are old. When a fuel treatment occurs, the fuel age of a cell is reset to zero. Also, fuel treatment activities have a cost and there is an available limited budget for treatments in each period. The problem calls for finding a suitable selection of the areas to be treated so as to minimize the presence of old contiguous areas over the whole time horizon. This problem can be typically represented by means of grid graphs where vertices represent areas and edges represent the connection between adjacent areas. Two areas may be connected not only in space but also according to other conditions such as prevailing wind directions.

In the following, we strongly refer to Minas et al. (2014), knowledge of which is assumed, with emphasis on the MILP model, the relevant applications, explanatory figures and problem parameter settings.

The main contributions of our work are summarized below.

- We first focus on the complexity status of BFTS on grid graphs, which is, to the authors' knowledge, still an open question, and we show that the decision version of BFTS on paths (and henceforth also on grid graphs) is strongly *NP*-complete. Then, we show that no polynomial time approximation algorithm exists for BFTS on paths unless $\mathcal{P} = \mathcal{NP}$.
- Next, we focus on the practical solvability of the problem. In the ILP formulations in Leon et al. (2019) and Minas et al. (2014), variables X_{it} indicate if area i is or not treated at time period t and variables O_{it} indicate if area i is or not old at time period t. In these formulations, however, big-M constraints are used. This may strongly limit the performance of MILP solvers in reaching optimal/suboptimal solutions. We show that a pair of alternative and much more simplified ILP formulations exists that avoids the need of big-M constraints. The best of these formulations when tested by means of MILP solver CPLEX 12.8 allows to optimally solve reasonably large size instances of the problem within limited CPU time.
- We consider also a new harder problem variant with periodic planning (hereafter denoted PBFTS) where the goal is to find a cyclic treatment plan with cycles of length T starting from an initial state to be determined. On this problem variant, CPLEX 12.8 applied to the proposed ILP starts running into difficulties with instances with 400 cells. Correspondingly, we propose a matheuristic approach in order to deal with larger size problems.
- We tested the effectiveness of the developed models and algorithms on a representative set of instances with up to 1225 cells.

The paper proceeds as follows. Section 2 provides the relevant notation for BFTS and discusses complexity and approximation issues. Section 3 presents enhanced ILP formulations both for BFTS and its periodic variant. Section 4 presents the proposed matheuristic approach and Section 5 provides the relevant related computational results both on BFTS and PBFTS. Section 6 concludes the paper with final remarks.

2. Notation and complexity analysis of BFTS

In BFTS, a set A of areas (or cells) to be treated and a time horizon T are given. For each area i, we denote by ϕ_i the set of its adjacent areas, by a_i its initial fuel age and by σ_i its fuel age threshold value. The threshold σ_i defines the number of periods, starting from the last treatment, after which area i becomes old. In each period $t = 1,\ldots,T$, there are treatment costs for each area i denoted by c_{it} and a budget on fuel treatments denoted by b_t. The length of the time horizon typically depends on the specific growth of the considered landscape and generally is between 5 and 15 years, according to the application of interest. In this work, we consider $T = 10$ as in previous relevant works from the literature. Yet, preliminary testing on other values of T in the range 5–15 showed that our models were not significantly affected in their performances.

Finally, we denote by w_{tij} the cost associated with the presence of two old adjacent areas i and j in period t. The problem calls for minimizing the weighted sum of old contiguous areas over the whole time horizon by a proper selection of the areas to treat without exceeding the available budget in each period.

BFTS can be modeled by means of grid graphs in some relevant applications. Typical representations of it are provided by the illustrative examples shown in Figs. 3–5 in Minas et al. (2014).

Here, we are interested in determining the computational complexity of the BFTS problem on this specific class of graphs or any related subclass. Let denote by BFTS-B the decision version of problem BFTS where we search for a feasible solution of BFTS so that the weighted sum of old contiguous areas does not exceed a given bound B within the whole time horizon T. We prove that BFTS-B is strongly *NP*-complete on paths (and consequently also on grid graphs) by reduction from the 3-PARTITION problem below which is well known to be *NP*-complete in the strong sense (Garey & Johnson, 1982).

3-PARTITION

INSTANCE: Positive integers n, K and a set of integers $S = \{s_1, \cdots, s_{3n}\}$ with $\sum_{i=1}^{3n} s_i = nK$ and $\frac{K}{4} < s_i < \frac{K}{2}$ for $i = 1, \cdots, 3n$.

QUESTION: Does there exist a partition $\{S_1, \cdots, S_n\}$ of S into 3-elements sets such that, for each $j, \sum_{s_i \in S_j} s_i = K$?

Theorem 1. *BFTS-B on paths is strongly NP-complete.*

Proof. We focus on a special case of problem BFTS-B where we search for a solution with no old contiguous areas, that is with $B = 0$. W.l.o.g we assume that all weights w_{tij} are one. Also, we have a path with $6n-1$ vertices and consider $T = n$. The path is constituted by $3n$ vertices $u_1, \cdots, u_{3n}$ and $3n-1$ vertices $v_1, \cdots, v_{3n-1}$ sequenced as follows:

$$u_1 - v_1 - u_2 - v_2 - \cdots - u_i - v_i - \cdots - u_{3n-1} - v_{3n-1} - u_{3n}.$$

All the $6n-1$ vertices have zero initial age and an identical threshold σ equal to $n-1$. All u_i vertices correspond to elements $s_i \in S$ of 3-PARTITION. These vertices have a treatment cost in each period equal to s_i. The budget in each period is $b = \frac{\sum_{i=1}^{3n} s_i}{n} = K$. All v_i vertices have cost $b + 1$, thus these vertices can never be treated. Hence, to find a feasible solution with no old contiguous areas, all u_i vertices must be treated in one of the periods $1, \cdots, n$ so that at period T when all v_i vertices are old, all u_i vertices are not old. Correspondingly, the sum of the treatment costs over the n periods is equal to the sum of all the s_i elements in 3-PARTITION, i.e. $\sum_{i=1}^{3n} s_i = nK$. Besides, the sum of the treatment costs in each period cannot exceed K. These conditions imply that this special case of BFTS-B has a solution if and only if 3-PARTITION has a solution: a solution of 3-PARTITION gives a solution of BFTS-B where the elements s_i in each subsets $S_1, \cdots, S_n$ indicate the associated vertices u_i to be treated in each period $1, \cdots, n$. Likewise, as $\frac{K}{4} < s_i < \frac{K}{2}$ for $i = 1, \cdots, 3n$, a solution of BFTS-B must treat exactly three vertices in each period with treatment costs equal to K, thus providing a solution of 3-PARTITION. □

In terms of approximability, the strong NP-completeness result of Theorem 1 rules out the existence of a fully polynomial time scheme for BFTS even on paths. However, we can prove a more general result that no polynomial time approximation algorithm exists for BFTS on paths unless $\mathcal{P} = \mathcal{NP}$.

We say that an approximation algorithm for BFTS has a finite approximation ratio $\rho \geqslant 1$ if it provides a solution value that is not larger than the product between ρ and the optimal solution value in any instance. Also, using the notation for 3-PARTITION, we recall the well known *NP*-complete PARTITION problem (Garey & Johnson, 1982) below.

PARTITION

INSTANCE: Finite set S of n positive integers $s_1, \cdots, s_n$.

QUESTION: Is there a subset $S' \subseteq S$ such that $\sum_{s_i \in S'} s_i = \sum_{s_i \in S \setminus S'} s_i = \frac{\sum_{i=1}^n s_i}{2}$?

By using a reasoning similar to the one in the proof of Theorem 1, we state the following theorem.

Theorem 2. *No polynomial time approximation algorithm exists for BFTS on paths unless $\mathcal{P} = \mathcal{NP}$.*

Proof. For a given instance of PARTITION, we consider an instance of BFTS with unit weights w_{tij} and $T = 2$. We consider a path with n vertices $u_1, \cdots, u_n$ and $n-1$ vertices $v_1, \cdots, v_{n-1}$ connected as follows:

$$u_1 - v_1 - u_2 - v_2 - \cdots - u_i - v_i - \cdots - u_{n-1} - v_{n-1} - u_n.$$

All the $2n-1$ vertices have zero initial age and a unit fuel threshold. All u_i vertices have a treatment cost equal to s_i in each period, thus they correspond to elements $s_i \in S$ of the instance of PARTITION. The budget in each period is $b = \frac{\sum_{i=1}^n s_i}{2}$. All v_i vertices have treatment cost $b+1$, so they cannot be treated and become old in the second period.

If the given instance of PARTITION has a solution, we can treat the vertices u_i associated with elements $s_i \in S'$ in the first period, the remaining vertices u_i in the second period (as in both periods the budget is $\frac{\sum_{i=1}^n s_i}{2}$) and obtain a solution for BFTS with no old contiguous areas and zero objective value.

Else, any optimal solution of the BFTS instance would induce at least two old contiguous areas, implying a positive objective function. Hence, a polynomial time approximation algorithm for BFTS would allow us to decide the PARTITION problem by checking if the approximate solution of BFTS is strictly positive. Obviously, this is not possible unless $\mathcal{P} = \mathcal{NP}$. □

Notice that the inapproximability result of Theorem 2 already applies for BFTS on paths with $T = 2$ and unit w_{tij}.

3. Enhanced ILP formulations for BFTS

We focus now on the practical solvability of BFTS. We introduce the following simplified ILP formulation that avoids the use of big-M constraints as in the model proposed in Minas et al. (2014) (Section 2). We consider binary variables x_{ti} equal to 1 iff area $i \in A$ is treated at time t and binary variables z_{tij} equal to 1 if and only if two adjacent areas i and j are old in period t. Correspondingly, we obtain the following model denoted as M_1^{BFTS}:

M_1^{BFTS} :

$$\text{minimize} \sum_{t=1}^{T} \sum_{i \in A} \sum_{j \in \phi_i} w_{tij} z_{tij} \quad (1)$$

subject to

$$\sum_{\substack{p = \max \\ \{1; t-\sigma_i\}}}^{t} x_{pi} + \sum_{\substack{p = \max \\ \{1; t-\sigma_j\}}}^{t} x_{pj} + z_{tij} \geqslant 1 \quad \forall i \in A, j \in \phi_i, t = 1, \dots, T : t > \max\{\sigma_i - a_i,\ \sigma_j - a_j\} \quad (2)$$

$$\sum_{i \in A} c_{ti} x_{ti} \leqslant b_t \quad t = 1, \dots, T \quad (3)$$

$$x_{ti} \in \{0, 1\} \quad \forall i \in A, t = 1, \dots, T \quad (4)$$

$$z_{tij} \in \{0, 1\} \quad \forall i \in A, j \in \phi_i, t = 1, \dots, T. \quad (5)$$

The objective function (1) minimizes the weighted sum of old adjacent areas over the whole time horizon. Constraints (2) detect the presence of old adjacent cells i, j in period t: if in the relevant periods both areas i and j are not treated (i.e. all the related variables x are equal to zero), then we have $z_{tij} = 1$; else we have $z_{tij} = 0$ given the objective function (1). Constraints (3) represent the budget constraints for each period $t = 1, \dots, T$. Constraints (4) and (5) define the domain of the variables. We point out that variables z_{tij} could be defined as nonnegative continuous variables, namely $z_{tij} \geqslant 0$, as variables z_{tij} are set to either 0 or 1 in any optimal solution due to constraints (2) and the objective function (1). Notice also that model M_1^{BFTS} can be seen as a variant of the set covering problem. This formulation considerably enhances the MILP model for BFTS proposed in Minas et al. (2014). As illustrated in our computational tests (see Section 5), a MILP solver launched on model M_1^{BFTS} already provides better performances than the reference model from the literature.

In model M_1^{BFTS}, we notice that, for any given area k, the sum $\sum_{p=\max\{1;t-\sigma_k\}}^{t} x_{pk}$ has to be repeated each time the area is considered in constraints (2). This may induce a large number of nonzero coefficients in the constraint matrix whenever the value of each threshold σ_i and the overall number of adjacent areas are not small. Generally speaking, a large number of nonzero coefficients may affect the performance of a MILP solver launched on model M_1^{BFTS}. Hence, we propose an alternative ILP formulation where we replace each sum $\sum_{p=\max\{1;t-\sigma_k\}}^{t} x_{pk}$ in constraints (2) with one auxiliary binary variable y_{tk} equal to 1 if area k is young at time t or equal to 0 if area k is old at time t. This condition is ensured by introducing an additional constraint $\sum_{p=\max\{1;t-\sigma_k\}}^{t} x_{pk} - y_{tk} \geqslant 0$. The constraint states if no treatment occurs for area k in the relevant periods up to t, then the area will be old, i.e. $y_{tk} = 0$. Correspondingly, we obtain the following model M_2^{BFTS}:

M_2^{BFTS} :

$$\text{minimize} \sum_{t=1}^{T} \sum_{i \in A} \sum_{j \in \phi_i} w_{tij} z_{tij} \quad (6)$$

subject to

$$y_{ti} + y_{tj} + z_{tij} \geqslant 1 \quad \forall i \in A, j \in \phi_i, t = 1, \dots, T : t > \max\{\sigma_i - a_i,\ \sigma_j - a_j\} \quad (7)$$

$$\sum_{\substack{p=\max \\ \{1;t-\sigma_i\}}}^{t} x_{pi} - y_{ti} \geqslant 0 \quad \forall i \in A, t = 1, \dots, T : t > \sigma_i - a_i \quad (8)$$

$$\sum_{i \in A} c_{ti} x_{ti} \leqslant b_t \quad t = 1, \dots, T \quad (9)$$

$$y_{ti}, x_{ti} \in \{0, 1\} \quad \forall i \in A, t = 1, \dots, T \quad (10)$$

$$z_{tij} \in \{0, 1\} \quad \forall i \in A, j \in \phi_i, t = 1, \dots, T. \quad (11)$$

The computational tests on reference instances from the literature (see Section 5) show that model M_2^{BFTS} provides better performances than model M_1^{BFTS}.

For the sake of exposition, from now on we refer only to model M_2^{BFTS} in the description of problem variants and of the matheuristic approach presented in Section 4. However, we remark that the same analysis can be easily extended to model M_1^{BFTS} as well.

3.1. Periodic BFTS

From a practical point of view, it might be of interest for the decisions makers to define periodic policies for fuel management. This motivates us to consider a new problem variant with periodic planning, here denoted as PBFTS. The goal is to find a continuously repeatable treatment plan with cycles of length T starting from an initial state to be determined. In this variant we assume that the initial ages a_i are set to zero. To handle periodicity, we replace constraints (8) in model M_2^{BFTS} with the following constraints:

$$\sum_{p=t-\sigma_i}^{t} x_{pi} - y_{ti} \geqslant 0 \quad \forall i \in A, t = 1, \dots, T : t - \sigma_i \geqslant 1 \tag{12}$$

$$\sum_{p=1}^{t} x_{pi} + \sum_{p=T+t-\sigma_i}^{T} x_{pi} - y_{ti} \geqslant 0 \quad \forall i \in A, t = 1, \dots, T : t - \sigma_i < 1. \tag{13}$$

and denote the corresponding model as M_2^{PBFTS}. Similarly to constraints (8), constraints (12) state that if $t > \sigma_i$, then the fact that area i is old or not in period t depends on the presence of treatments in periods $t-\sigma_i, \dots, t$. If instead $t \leqslant \sigma_i$, also periods from the preceding cycle have to be considered according to constraints (13).

4. A matheuristic algorithm

Computational tests on problem PBFTS (see Section 5) show that the addition of periodicity makes the problem much more challenging so that already for several instances with 100 cells, the considered ILP solver (CPLEX 12.8) applied to model M_2^{PBFTS} fails to reach an optimal solution within 1800 s of CPU time, denoting the need of a heuristic algorithm: to this extent, we propose a matheuristic approach.

Matheuristics are methods that attracted the attention of the community of researchers (see for instance Ball, 2011; Della Croce, Grosso, & Salassa, 2013), giving rise to an impressive amount of research in recent years. We mention applications of matheuristics on routing (Macrina, Laporte, Guerriero, & Di Puglia Pugliese, 2019), packing (Martinez-Sykora, Alvarez-Valdes, Bennell, Ruiz, & Tamarit, 2017), rostering (Doi, Nishi, & Voss, 2018) and machine scheduling (Della Croce, Grosso, & Salassa, 2014) just to cite a few of them. Matheuristics rely on the general idea of exploiting the strength of both metaheuristic algorithms and exact methods. Here, the proposed matheuristic algorithm is based on an overarching neighborhood search approach with an intensification search phase realized by a MILP solver. The proposed MILP formulation for BFTS and PBFTS constitutes a backbone model generalizable also to the variants presented in Leon et al. (2019), Minas et al. (2014), Rachmawati et al. (2018) and Rachmawati et al. (2016). We cite, among others, the possible request of having no adjacent areas treatment in the same time period and minimal or maximal periods of time between two treatments of the same areas.

The proposed matheuristic algorithm relies on computing an initial solution from the optimal solution of the linear relaxation of M_2^{PBFTS}, denoted as $M_2^{PBFTS-REL}$. Then, a local search procedure is applied to improve the incumbent solution. Both the initial solution and the improvement procedure can be also applied to BTFS without major modifications: what changes is just the underlying model used by the algorithms.

The main idea of the initial solution procedure (denoted as Algorithm 1) is to handle $M_2^{PBFTS-REL}$ by iteratively setting to 0 or 1 variables x_{it}, until all of them have been fixed. The algorithm begins from the first time period and sets at most K fractional variables (the ones with the highest value in the current optimal solution), without exceeding the budget of the period. After, $M_2^{PBFTS-REL}$ is solved again until all variables are integer. Then, we move to the next time period and the same procedure applies.

The detailed steps of the procedure are summarized as follows. The inputs (Line 1) are value K and model $M_2^{PBFTS-REL}$. At first, the current time period is $t_{curr} = 1$, the budget $B_t = b_t$ for each time period t, and the sets Φ_0 and Φ_1 of indices (t, i) of the x_{it} variables set to 0 or 1 are empty (Line 2). Then, we solve the linear relaxation of the model, retrieving the initial solution $\overline{X}$ (Line 3). The main cycle (Lines 4–28) is iterated until $\overline{X}$ becomes integer. In Lines 5–13, we fix to 0 or 1 any (not yet constrained) variable $\overline{x}_{it}$ assuming an integer value in solution $\overline{X}$. Any time a variable is set to 1, the remaining budget of the period is updated accordingly (Line 11). In Lines 14–26, we fix to 1 (at most) the largest K variables x_{it}, with $t = t_{curr}$. In particular, variable x_{it} can be fixed to 1 only if there is sufficient remaining budget in the current time period (Lines 20–22). All variables whose cost exceeds the current budget are set to 0 (Lines 23–24). The cycle ends when all variables of time period t_{curr} have been inserted in Φ_0 or Φ_1 (Lines 16–18). Then, model $M_2^{PBFTS-REL}$ is solved again with the additional constraints on the fixed variables (Line 27) and the process is iterated.

Algorithm 1. Initial solution.

1: INPUT: The maximum number of variables fixed at each iteration K, model $M_2^{PBFTS-REL}$.
2: Set $t_{curr} = 1; \Phi_0 = \Phi_1 = \emptyset; B_t = b_t, t \in 1, \cdots, T$
3: Solve $M_2^{PBFTS-REL}$. Let $\overline{X}$ be the solution.
4: **while** solution $\overline{X}$ is not integer **do**
5: **for all** (t, i) indices such that $t \geqslant t_{curr}; (t,i) \notin \Phi_0; (t,i) \notin \Phi_1$ **do**
6: **if** $\overline{x}_{ti} = 0$ **then**
7: $\Phi_0 = \Phi_0 + (t, i)$
8: **end if**
9: **if** $\overline{x}_{ti} = 1$ **then**
10: $\Phi_1 = \Phi_1 + (t, i)$
11: $B_t = B_t - c_{ti}$
12: **end if**
13: **end for**
14: **for** $count = 1, \cdots, K$ **do**
15: Let i be the index corresponding to the highest $\overline{x}_{t_{curr},i}$ such that $(t_{curr}, i) \notin \Phi_0$, $(t_{curr}, i) \notin \Phi_1$.
16: **if** there is no such an index **then**
17: $t_{curr} = t_{curr} + 1;$
18: **break for**
19: **end if**
20: **if** $B_{t_{curr}} \geqslant c_{t_{curr},i}$ **then**
21: $\Phi_1 = \Phi_1 + (t_{curr}, i)$
22: $B_{t_{curr}} = B_{t_{curr}} - c_{t_{curr},i}$
23: **else**
24: $\Phi_0 = \Phi_0 + (t_{curr}, i)$
25: **end if**
26: **end for**
27: Solve again $M_2^{PBFTS-REL}$, adding constraints $x_{ti} = 0$ for all $(t,i) \in \Phi_0$ and $x_{ti} = 1$ for all $(t,i) \in \Phi_1$. Let again $\overline{X}$ be its optimal solution.
28: **end while**
29: OUTPUT: the feasible (integer) solution $\overline{X}$.

The matheuristic improvement procedure (denoted as Algorithm 2) starts with the solution found by Algorithm 1 and iteratively improves the solution with a neighborhood search approach. Each iteration explores the neighborhood by considering a subproblem where all variables related to a subset of time periods have to be optimized, while the remaining variables are fixed to the value they have in the current solution. The neighborhood size is variable, beginning with a smaller number of periods, which is increased when a complete exploration of the neighborhood has been performed without finding any improvement.

More in details, the procedure requires an integer solution $\overline{X}$ and the initial and final neighborhood size T_{init} and T_{final} (Line 1). The main cycle (Lines 3–21) reoptimizes at each iteration a subset of the current solution, creating a subproblem with a consecutive number of $T_{dim} < T$ periods, with T_{dim} ranging from T_{init} to T_{final}. The cycle begins setting $t_{min} =$ 1and $t_{impr} = T$ (Line 3), where t_{min} is the initial period to be reoptimized and t_{impr} is the period when the neighborhood exploration should end. Then, an internal cycle is repeated while $t_{min} \neq t_{impr}$ (Lines 4–20). Here, all the x_{it} variables, with t in the interval $[t_{min}, t_{min} + T_{dim} - 1]$, are the variables to be reoptimized. Notice that, when the second limit exceeds T, the reoptimization interval resumes from time 1 implementing a cycling neighborhood, thus becoming $[t_{min}, T] \cup [1, t_{min} + T_{dim} - T - 1]$. All variables not belonging to the reoptimization interval are fixed to the value they already have in the current solution (Lines 5–10). The model is then solved to optimality, where, if the new solution $\overline{X}$ (with objective value $OF(\overline{X})$) is better than the previous one, t_{impr} is updated (Lines 11–15). Note that at each iteration the optimal solution of the reoptimization problem cannot be worse than the current solution (the latter still being a feasible solution for the new model). Then, a new neighborhood is obtained by increasing t_{min}, or by setting the parameter back to 1 if T has been reached (Lines 16–19).

Algorithm 2. Matheuristic Improvement Procedure.

1: INPUT: an initial solution $\overline{X}$, initial and final neighborhood size T_{init} and T_{final}, model $ILP = M_2^{PBFTS}$.
2: **for** $T_{dim} = T_{init}, \cdots, T_{final}$ **do**
3: Set $t_{min} = 1, t_{impr} = T$ **do**
4: **while** $t_{min} \neq t_{impr}$ **do**
5: Consider model ILP (without any additional constraint)
6: **if** $t_{min} + T_{dim} - 1 \leqslant T$ **then**
7: Add to ILP constraints $x_{ti} = \overline{x}_{ti}$ for all t such that $t < t_{min}$ or $t > t_{min} + T_{dim} - 1$
8: **else**
9: Add to ILP constraints $x_{ti} = \overline{x}_{ti}$ for all t such that $t_{min} + T_{dim} - T - 1 < t < t_{min}$
10: **end if**
11: $O_{last} = OF(\overline{X})$
12: Optimally solve model ILP, retrieving the new solution $\overline{X}$
13: **if** $OF(\overline{X}) < O_{last}$ **then**
14: $t_{impr} = t_{min}$
15: **end if**
16: $t_{min} = t_{min} + 1$
17: **if** $t_{min} > T$ **then**
18: $t_{min} = 1$
19: **end if**
20: **end while**
21: **end for**
22: OUTPUT: the improved solution $\overline{X}$.

5. Computational results

We focused on grid graphs and generated instances according to the scheme proposed in Minas et al. (2014) that we describe in the following. We considered graphs with 25, 100, 225, 400, 900 and 1225 cells where each fuel age threshold σ_i is randomly selected among 4, 8, or 12 years and each fuel age a_i is randomly selected between 1 and 12 years. The time horizon T is 10 years. Each cell is connected with three neighboring cells by considering north-westerly prevailing wind direction, as indicated in Minas et al. (2014). Objective function weights and cells treatment costs are constant ($w_{tij} = c_{ti} = 1$ for all $t = 1, \cdots, T$ and $i \in A$). Each budget value b_t is equal to 5% of the total treatment cost of all the cells, as considered in the most difficult instances in Minas et al. (2014). A second instance type has been then generated, considering both weights and treatment costs as random integer numbers uniformly distributed in $[1, 20]$. For each of the two instance types and 6 landscape size we generated 10 instances, for a total of 120 different instances. All tests have been run using MILP solver CPLEX 12.8 running on an Intel i5 CPU @ 3.0 GHz with 16 GB of RAM, within a time limit of 1800 s.

In the first computational tests, we considered for the non periodic version of the problem (BFTS) the model in Minas et al. (2014) and our models M_1^{BFTS} and M_2^{BFTS}. In Table 1, we report the performances of the models on the instances with constant costs and weights. For each landscape size, we report the average solution value (column "Average Sol. Value"), the average CPU time in seconds (column "Average Time"), and the number of instances, out of 10, solved to optimality with each model (column "Opt"). In all tables, bold entries highlight the best average solution value for each landscape size. The results illustrate that the model proposed in Minas et al. (2014) is not capable of solving to optimality any of the instances with 400 or more cells. Model M_1^{BFTS} is able to solve to optimality all the instances up to 400 nodes, while model M_2^{BFTS} exhibits much better performances and reaches all optimal solutions within the time limit. A possible explanation on the improved performances of the latter model is that in all instances the number of nonzero coefficients in model M_2^{BFTS} is more than halved (on average) with respect to model M_1^{BFTS}.

The same tests have been repeated for the instances with variable costs and weights. The results are summarized in Table 2. The model proposed in Minas et al. (2014) (adapted by adding weights w_{tij} in the objective function) is capable of solving to optimality all the instances only for the landscapes with 25 and 100 cells. The proposed models are able to solve all the instances up to 225 cells. Again, the best performing model is M_2^{BFTS}, able to solve all instances but 3 in the given time limit (one each for the 400, 900 and 1225 cells sets). Notice also that in the latter instances model M_2^{BFTS} computes almost optimal solutions. The optimality gap $1 - \frac{LB}{UB}$ between the solution value UB and the lower bound LB computed within the time limit is always very low (respectively equal to 0.01%, 0.03%, 0.01% in the instances with 400, 900 and 1225 cells). Considering that 117 out of 120 instances are already optimally solved by the model, the matheuristic has not been tested on the non periodic case.

We then considered the same 120 instances in the context of periodic planning and benchmarked the proposed matheuristic against model M_2^{PBFTS}. The CPU time limit was again 1800 s. The matheuristic have been tested with several configurations: in the procedure for the initial solution we tested K ranging from 1 to 100, while in the improvement procedure the interval $T_{init} - T_{final}$ was ranging, with different combinations, from 2 to 6. The results presented here refer to the settings $K = 20$, $T_{init} = 4$, $T_{end} = 5$, which guarantee a good trade-off between running times and solution quality.

The results for the instances with constant costs and weights are reported in Table 3. The table also includes the average optimality gaps in percentage value (column "Average Gap (%)"), where we consider the LB values computed by model M_2^{PBFTS}. In general, the presence of a periodic planning makes the instances harder to solve. In fact, the proposed model does not obtain all optimal solutions and is outperformed, in terms of solution quality, by the matheuristic on large instances with 400, 900 and 1225 cells. Notice that the matheuristic considerably reduces the optimality gaps of model M_2^{PBFTS} (by more than 10% on average) in the instances with 900 and 1225 cells.

As in Table 3, we present in Table 4 the results for the periodic planning on the instances with variable costs and weights. Again, the model is not able to optimally solve the largest instances within the time

Table 1
BFTS instances with constant costs and weights.

	Model in Minas et al. (2014)			Model M_1^{BFTS}			Model M_2^{BFTS}		
Landscape Size	Average Sol. Value	Average Time	Opt	Average Sol. Value	Average Time	Opt	Average Sol. Value	Average Time	Opt
25 (5 by 5)	**113.7**	1.2	10	**113.7**	< 0.1	10	**113.7**	< 0.1	10
100 (10 by 10)	**360.2**	40.9	10	**360.2**	3.1	10	**360.2**	1.3	10
225 (15 by 15)	925.7	770.5	7	**925.7**	68.7	10	**925.7**	7.3	10
400 (20 by 20)	1,645.6	1,800.0	0	**1,640.5**	362.7	10	**1,640.5**	38.2	10
900 (30 by 30)	4,766.8	1,800.0	0	3,936.1	1,800.0	0	**3,809.9**	282.4	10
1,225 (35 by 35)	6,811.5	1,800.0	0	6,027.7	1,800.0	0	**5,257.4**	867.9	10

Table 2
BFTS instances with variable costs and weights.

	Model in Minas et al. (2014)			Model M_1^{BFTS}			Model M_2^{BFTS}		
Landscape Size	Average Sol. Value	Average Time	Opt	Average Sol. Value	Average Time	Opt	Average Sol. Value	Average Time	Opt
25 (5 by 5)	**590.2**	< 0.1	10	**590.2**	< 0.1	10	**590.2**	< 0.1	10
100 (10 by 10)	**1,468.0**	270.2	10	**1,468.0**	22.9	10	**1,468.0**	27.7	10
225 (15 by 15)	3,323.3	1,539.0	3	**3,320.4**	189.5	10	**3,320.4**	219.4	10
400 (20 by 20)	6,191.5	1,800.0	0	**6,175.1**	885.7	8	**6,175.1**	695.0	9
900 (30 by 30)	20,430.9	1,800.0	0	14,756.9	1,193.5	5	**14,756.1**	1,041.8	9
1,225 (35 by 35)	32,824.4	1,800.0	0	21,027.3	1,424.4	5	**21,026.5**	1,104.8	9

Table 3
PBFTS instances with constant costs and weights.

	Model M_2^{PBFTS}				Matheuristic algorithm		
Landscape Size	Average Sol. Value	Average Time	Average Gap (%)	Opt	Average Sol. Value	Average Time	Average Gap (%)
25 (5 by 5)	**166.1**	0.5	0.00	10	166.5	0.1	0.22
100 (10 by 10)	**519.5**	916.0	0.15	7	520.2	6.4	0.29
225 (15 by 15)	**1,340.2**	1,800.0	0.69	0	1,342.2	20.9	0.84
400 (20 by 20)	2,347.9	1,800.0	1.08	0	**2,347.4**	64.9	1.07
900 (30 by 30)	7,151.8	1,800.0	15.37	0	**5,426.6**	360.6	1.37
1,225 (35 by 35)	9,035.3	1,800.0	11.24	0	**7,483.9**	775.3	1.20

Table 4
PBFTS instances with variable costs and weights.

	Model M_2^{PBFTS}				Matheuristic algorithm		
Landscape	Average Sol. Value	Average Time	Average Gap (%)	Opt	Average Sol. Value	Average Time	Average Gap (%)
25 (5 by 5)	**849.4**	0.1	0.00	10	850.1	0.2	0.10
100 (10 by 10)	**2,399.1**	1,052.5	0.21	5	2,409.9	9.7	0.66
225 (15 by 15)	**5,754.9**	1,647.1	0.70	1	5,769.1	38	0.96
400 (20 by 20)	**10,822.8**	1,800.0	1.23	0	10,853.4	105.1	1.50
900 (30 by 30)	25,347.8	1,800.0	3.91	0	**25,077.5**	490.9	2.88
1,225 (35 by 35)	45,109.6	1,800.0	12.14	0	**35,241.5**	1,043.8	2.51

limit, and the matheuristic algorithm outperforms it in terms of solution quality for the largest landscapes (900 and 1225 cells).

6. Conclusions and future directions

In this work, we proposed new mathematical models and a matheuristic approach for the Budget constrained Fuel Treatment Scheduling Problem, a well-known optimization problem in fuel management. We evaluated the effectiveness of the developed models and algorithms on a large set of instances from the literature and for a problem variant with periodic planning. In our algorithmic developments, we also investigated a tailored column generation approach. We decomposed model M_2^{BFTS} by considering all possible feasible treatment plans of the areas in each period $t = 1,\dots,T$. This induced a number of plans that was possibly exponential in the number of areas. Correspondingly, for each period t we defined a pricing problem to compute feasible treatment plans. The choice of treatment plan s in period t was associated with a binary variable α_s^t in a master problem containing also variables y_{ti}, z_{tij} and without variables x_{ti}. However, computational tests on small instances showed that solving the linear relaxation of the master problem did not require negligible computational times due to the presence of variables y_{ti} and z_{tij}. Thus, the use of such a column generation approach does not seem a viable option for BFTS. In future research, it would be worthy to consider further generalizations of the problem, in order to identify better fits with real-life scenarios. In this respect, extensions of the ILP models proposed in Minas et al. (2014) were considered in Leon et al. (2019), Rachmawati et al. (2018) and Rachmawati et al. (2016). Interestingly, our approaches can be easily adapted to most of these extensions as mentioned in the Introduction. In addition, requirements to preserve the fauna habitat could be taken into account in deriving fuel management strategies. For instance, an area can be treated in a given period only if the present fauna could move to a sufficient number of adjacent areas with a suitable habitat. Finally, here we sticked to the use of landscapes divided into a grid of square cells as assumed in the work of Minas et al. (2014) but our approach is extendable to any landscape that can fit into a network representation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was funded by the GEO-SAFE project and the EU Horizon2020 RISE programme, grant agreement No. 691161.

References

Ball, M. O. (2011). Heuristics based on mathematical programming. *Surveys in Operation Research and Management Science, 16*, 21–38.
Boer, M. M., Sadler, R. J., Wittkuhn, R. S., McCaw, L., & Grierson, P. F. (2009). Long-term impacts of prescribed burning on regional extent and incidence of wildfires-evidence from 50 years of active fire management in sw australian forests. *Forest Ecology and Management, 259*(1), 132–142.
Della Croce, F., Grosso, A., & Salassa, F. (2013). Matheuristics: embedding MILP solvers into heuristic algorithms for combinatorial optimization problems. In P. Siarry (Ed.), *Heuristics: theory and applications* (pp. 31–52). Nova Science Publishers.
Della Croce, F., Grosso, A., & Salassa, F. (2014). A matheuristic approach for the two-machine total completion time flow shop problem. *Annals of Operations Research, 213*(1), 67–78.

Demange, M., & Tanasescu, C. (2015). A Graph Approach for Fuel Treatment Scheduling. Working paper. RMIT University.

Doi, T., Nishi, T., & Voss, S. (2018). Two-level decomposition-based matheuristic for airline crew rostering problems with fair working time. *European Journal of Operational Research, 267*(2), 428–438.

Fernandes, P. M., & Botelho, H. S. (2003). A review of prescribed burning effectiveness in fire hazard reduction. *International Journal of Wildland Fire, 12*(2), 117–128.

Garey, M. R., & Johnson, D. S. (1982). *Computers and Intactability: A Guide to the Theory of NP-Completeness*. New York: Freeman.

Leon, J., Reijnders, V. M. J. J., Hearne, J. W., Ozlen, M., & Reinke, K. J. (2019). A Landscape-Scale Optimisation Model to Break the Hazardous Fuel Continuum While Maintaining Habitat Quality. *Environmental Modelling and Assessment, 24*, 369–379.

Mackenzie, B. D. E., Clarke, S. W., Zimmer, H. C., Liew, E. C. Y., Phelan, M. T., Offord, C. A., … Auld, T. D. (2021). Ecology and Conservation of a Living Fossil: Australia's Wollemi Pine (Wollemia nobilis). In *Reference Module in Earth Systems and Environmental Sciences*. Elsevier.

Macrina, G., Laporte, G., Guerriero, F., & Di Puglia Pugliese, L. (2019). An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time windows. *European Journal of Operational Research, 276*(3), 971–982.

Martinez-Sykora, A., Alvarez-Valdes, R., Bennell, J. A., Ruiz, R., & Tamarit, J. M. (2017). Matheuristics for the irregular bin packing problem with free rotations. *European Journal of Operational Research, 258*(2), 440–455.

Minas, J. P., Hearne, J. W., & Handmer, J. W. (2012). A review of operations research methods applicable to wildfire management. *International Journal of Wildland Fire, 21*(3), 189–196.

Minas, J. P., Hearne, J. W., & Martell, D. L. (2014). A spatial optimisation model for multi-period landscape level fuel management to mitigate wild reimpacts. *European Journal of Operational Research, 232*, 412–422.

Rachmawati, R., Ozlen, M., Hearne, J. W., & Reinke, K. J. (2018). Fuel treatment planning: Fragmenting high fuel load areas while maintaining availability and connectivity of faunal habitat. *Applied Mathematical Modelling, 54*, 298–310.

Rachmawati, R., Ozlen, M., Reinke, K. J., & Hearne, J. W. (2016). An optimisation approach for fuel treatment planning to break the connectivity of high-risk regions. *Forest Ecology and Management, 368*, 94–104.

Stephens, S. L., & Ruth, L. W. (2005). Federal Forest-Fire Policy in the United States. *Ecological Applications, 15*(2), 532–542.

Wei, Y., Rideout, D., & Kirsch, A. (2008). An optimization model for locating fuel treatments across a landscape to reduce expected fire losses. *Canadian Journal of Forest Research, 38*(4), 868–877.