A Study of Evacuation Planning for Wildfires

Christian Artigues[1], Emmanuel Hebrard[1], Yannick Pencole[1], Andreas Schutt[2,3], and Peter Stuckey[2,3]

[1] LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
[2] Decision Sciences, Data61 CSIRO, Melbourne, Australia
[3] Department of Computing and Information Systems, The University of Melbourne, Melbourne, Australia

Abstract. The GEO-SAFE project gathers academic researchers and fire emergency practitioners from EU and Australia with the objective to design innovative models and efficient response tools based on optimization methods for fighting wildfires. In this paper, we consider an evacuation planning problem in a scenario issued from models discussed with practitioners, where a wildfire is threatening a region with intermediate populated centres.
As in previous approaches proposed in case of flood, we use a constraint optimization model involving tasks to represent the evacuation of a population and a cumulative constraint per shared route segments. However, we consider malleable tasks. Indeed, in order to mitigate congestion risks, the authorities may delay the start of the evacuation but they may also affect the rate of evacuation by modulating the method used to alert the population. We also consider a different objective: we maximize the minimum "safety margin", weighted by the size of population, over every evacuated population and every road segment.
We introduce a new heuristic and a global flow constraint propagator. Moreover, we also propose an instance generator based on a random generation of road networks and basic fire propagation models. This generator produces challenging benchmarks even with very few evacuation tasks. Finally, we report the results of extensive computational experiments done using CP Optimizer, and discuss their relevance both to the GEO-SAFE project and to CP.

1 Introduction

In EU and Australia, every year thousands of square miles of forests and other lands burn due to wildfires. These fires cause important economic and ecological losses, and often, human casualties as for instance during the Black Saturday bushfires across the Australian state of Victoria in February 2009 [12]. The overall objective of the GEOSAFE project [7] is to develop methods and tools enabling to set up an integrated decision support system to assist authorities in optimizing the resources during the response phase to a wildfire (fire suppression, life and goods protection). One critical and crucial part of this integrated decision support is the ability to perform large-scale evacuation planning. As detailed in [13], there are commonly three categories of evacuees: the ones that leave early, the ones that shelter in refuge and the ones who stay at their properties and fight. This work focuses on the evacuation of this third group (late evacuees) which is called the *late evacuation planning problem*.

While in practice most evacuation planning is principally designed by experts using simple heuristic approaches or scenario simulations [15], more recently optimization approaches to evacuation planning have been addressed, using a variety of optimization technology [2]. The usual test case for most of this work is flood evacuation planning [10, 5, 8]. Given accurate measurements of rainfall and topology, flood evacuation planning can make use of very accurate predictions of future water levels, and therefore has a very accurate model of what infrastructure will be available at each stage of the evacuation.

Evacuation planning in case of wildfires is much harder. Wildfire propagations are inherently less predictable than floods. While flood levels mostly rely on the fixed topology of the area and rainfalls, wildfire mainly depends on the wildland fuels [11, 1], on the slope of the burning ground and more importantly on the speed and direction of the wind that can suddenly change at any time [16, 14]. Therefore, evacuation planning dedicated to wildfires must be much more robust to difference future scenarios. A good evacuation plan in case of wildfire must not only minimize the evacuation time of the population but also maximize the spatial and temporal safety margin between the evacuees and the actual or potential wildfire front.

In this paper, we consider that the authorities already identified the accessible routes and shelters and estimated the population of the late evacuees. Evacuation takes place in individual vehicles and each center has a known population and a single predefined suggested evacuation route. All routes issued from each center converge toward a safe place, so congestion may appear on route segment shared by several evacuation paths. The interest of convergent evacuation plans has been underlined by several studies such as in [4] since it avoids congestion issued by driver slow-downs at forks. Furthermore fire propagation models give a deadline on each route segment beyond which taking this segment comes at high risk. To mitigate these risks, the authorities may delay the evacuation start time and rate for each center. Indeed in practice, besides the possibility of assigning a start time to each evacuated zone, the authorities can also mobilize different levels of resources to increase the evacuation rate (e.g. number of agents knocking on people's door), to which people answer according to a behavioral model abstracted by a response curve [9]. Based on these concepts, the evacuation planning model is close to the one proposed in [5] called the non-preemptive evacuation planning problem (NEPP) in the context of a flood. In contrast with previously proposed models, the NEPP considers non preemptive evacuation. Once the evacuation of a zone has started, it cannot be interrupted. Indeed, considering preemptive evacuation would make the problem much easier to solve but is hard to implement in practice. In case of wildfire, this would notably cause undesirable stress on evacuees. The major difference of the model considered in this paper with the NEPP model lies in the objective function. In [5], the main objective was to maximize the number of evacuees, while a secondary objective was to minimize the makespan, i.e. the total evacuation time, while enforcing deadlines on route segments. In case of wildfire, the high variability in the fire propagation makes it necessary to avoid taking a route segment close to the expected deadline. Hence, we consider a single objective by maximizing the minimum gap for all evacuated zones and all route segments between the deadline and the by which the last evacuee leaves a route segment, weighted by the population of the evacuated zone.

2 The evacuation planning problem

We are given an a tree $\mathcal{G} = (\mathcal{E} \cup \mathcal{T} \cup \{r\}, \mathcal{A})$ rooted in r standing for the evacuation routes from *evacuation nodes* $\mathcal{E}$ (leaves) to the *safe node* r through *transit nodes* $\mathcal{T}$. Every leaf/evacuation node $v \in \mathcal{E}$ is associated with a population count w_v. Every arc u, u' has a length $l_{uu'}$ and a capacity non-ambiguously denoted q_u as $\mathcal{G}$ is a tree. Moreover the length of the path from a node u to a node u' is also written $l_{uu'}$.

Let $\mathcal{H} = [0, H]$ denote the time horizon of the evacuation planning. We want to associate every leaf/evacuation node $v \in \mathcal{E}$ to a real s_v representing the delay of the evacuation notice and to a "response curve" ϕ_v describing the evacuation flow out of node v over time (starting from s_v). Population flows out of leaf/evacuation node v at rate $\phi_v(t)$ that is zero before time $s_v \geq 0$, and such that:

$$\int_0^H \phi_v(t)dt = w_v$$

Let $p(u)$ denote the parent of u in $\mathcal{G}$, $\hat{p}(u)$ its ascendants, $C(u)$ its children, $\hat{C}(u)$ its descendants and $L(u)$ those of its descendants that are leaves of $\mathcal{G}$. We assume that the evacuees never stop, so the flow in the downstream arcs can be computed by summing the flows in the incoming arcs and the flow $\phi_u(t)$ for any arc $\in \mathcal{E} \cup \mathcal{T}$ at any time $t \in \mathcal{H}$ is

$$\phi_u(t) = \sum_{v \in L(u)} \phi_v(t - l_{uv} - s_v)$$

In this paper we consider as in [5] a simple response curve. The flow out of an evacuation node $v \in \mathcal{E}$ is a continuous decision variable and remain constant during the evacuation process, that is, the flow out of node $v \in \mathcal{E}$ is equal to h_v within the time interval $[s_v, e_v]$, with $e_v = s_v + \frac{w_v}{h_v}$ and zero otherwise. Therefore, the flow out of a node $u \in \mathcal{E} \cup \mathcal{T}$ at time $t \in \mathcal{H}$ is:

$$\phi_u(t) = \sum_{v \in L(u), s_v + l_{uv} \leq t < s_v + l_{uv} + \frac{w_v}{h_v}} h_v$$

It follows that the considered evacuation planning problem can be formally defined as the following constraint optimization problem.

Variables: We have a set of $|\mathcal{E}|$ non-preemptive tasks, one for every evacuation node. For every task standing for an evacuation node $v \in \mathcal{E}$ we need two variables, one for the constant rate $h_v \in]0, q_v]$ at which the evacuation will proceed and one for its start time $s_v \in [0, H - \frac{w_v}{q_v}]$ at which the evacuation will start.

Constraints: We have a single family of constraints: we must avoid "jams", that is, flow exceeding the capacity of an arc. More precisely, for each node $u \in \mathcal{T}$, we have a cumulative resource constraint of capacity q_u ensuring that $\phi_u(t) \leq q_u$, whis is written:

$$\sum_{v \in L(u), s_v + l_{uv} \leq t < s_v + l_{uv} + \frac{w_v}{h_v}} h_v \leq q_u, \quad \forall u \in \mathcal{T}, \forall t \in \mathcal{H}$$

Objective: To every transit node u is associated a due date d_u, corresponding to the time at which the following road portion becomes unsafe. The objective is to minimize the maximum lateness of any task, i.e., the difference between the time at which it leaves a node u and d_u, weighted by the population. Hence the objective is:

$$\min \max_{u \in \mathcal{T}, v \in L(u)} s_v + h_v/w_v + l_{uv} - d_u$$

Dominance rules and problem formulation A first observation is that we can simplify the objective function by retaining the transit node u minimizing $d_u - l_{uv}$. Let $d_v = \min_{u \in \mathcal{T}}\{d_u - l_{uv}\}$ denote this value.

The objective can therefore be rewritten as the maximum of $|\mathcal{E}|$ expressions:

$$\min \max_{v \in \mathcal{E}} s_v + h_v/w_v - d_v$$

The second following observation, also made in [5], allows to reduce the number of cumulative constraints to consider.

Observations 1 *For any two transit nodes $u, u' \in \mathcal{T}$, if $q_u \leq q_{u'}$ and $u' \in \hat{C}(u)$, then a jam in u' entails a jam in u.*

We thus only need to check jams in nodes u such that $\forall v \in \hat{p}(u),\ q_v > q_u$. In practice, it means that given an evacuation node $v \in \mathcal{E}$ we can explore the nodes in the route from v to r in reverse order, and for every stretch of road without branch, keep only the arc of minimal capacity, called the critical arc. Let $\tilde{\mathcal{T}}$ denote the reduced set of critical transit nodes to consider. The problem can thus be formulated as follows.

$$\text{minimize} \quad \max_{v \in \mathcal{E}} w_v\{s_v + h_v/w_v - d_v\} \tag{1}$$

$$\text{subject to:} \quad \sum_{v \in L(u), s_v + l_{uv} \leq t < s_v + l_{uv} + \frac{w_v}{h_v}} h_v \leq q_u, \quad \forall u \in \tilde{\mathcal{T}}, \forall t \in \mathcal{H} \tag{2}$$

$$h_v \in]0, q_v], s_v \in [0, H - \frac{w_v}{q_v}], \quad \forall v \in \mathcal{E} \tag{3}$$

3 Baseline approach using `cumulative` constraints

In [5], the NEPP was model using standard `cumulative` constraints. We consider this model for our problem as the baseline approach. Given a set of tasks J with start time $s_i \in [r_i, \infty]$, completion time $e_i \in [0, d_i]$, height $h_i \in [0, m_i]$ and a resource r of constant capacity q_r, $\texttt{cumulative}((s_i, e_i, h_i)_{i \in J}, q_r)$ enforces the relations

$$\sum_{i \in J | s_i \leq t \leq e_i} h_i \leq q_r \quad \forall t \in \mathcal{H} \tag{4}$$

Consequently to model the problem, it suffices to consider a resource-unconstrained task v for each evacuation node $v \in \mathcal{E}$ with height h_v, release date $r_v = 0$, due date d_v, and to duplicate and translate this task for each critical transit node u on its path towards the safe node. For each critical arc $u \in \tilde{\mathcal{T}}$, let i_{uv} denote the duplicate for evacuation node $v \in L(u)$. The duplicate has a release date $r_{i_{uv}} = l_{uv}$, a due date $d_{i_{uv}} = d_v + l_{uv}$ and a height $h_{i_{uv}} = h_v$. A resource is defined per critical arc $u \in \tilde{\mathcal{T}}$, with capacity q_u.

The baseline constraint program for the evacuation planning problem is obtained by replacing constraints (2) in the problem formulation by:

$$\texttt{cumulative}((s_{i_{uv}}, e_{i_{uv}}, h_{i_{uv}}), q_u) \qquad \forall u \in \tilde{\mathcal{T}} \tag{5}$$

$$w_v = h_v(e_v - s_v) \qquad \forall v \in \mathcal{E} \tag{6}$$

$$s_{i_{uv}} = s_v + l_{uv} \qquad \forall u \in \tilde{\mathcal{T}}, \forall v \in L(u) \tag{7}$$

$$e_{i_{uv}} = e_v + l_{uv} \qquad \forall u \in \tilde{\mathcal{T}}, \forall v \in L(u) \tag{8}$$

$$h_{i_{uv}} = h_v \qquad \forall u \in \tilde{\mathcal{T}}, \forall v \in L(u) \tag{9}$$

In the baseline model above, the information that tasks have a fixed energy is lost to the `cumulative` constraint propagator. Given a task v with energy w_v start time s_v, completion time e_v and rate h_v, the algorithm will consider a task of duration $\left\lceil \frac{w_v}{m_v} \right\rceil$ and consumption $\left\lceil \frac{w_v}{d_v - r_v} \right\rceil$. When the upper bounds on consumption and duration are large, these values tend to 0, thus greatly hindering constraint propagation.

Example 1. Consider a resource of capacity 4 and four tasks with the following parameters:

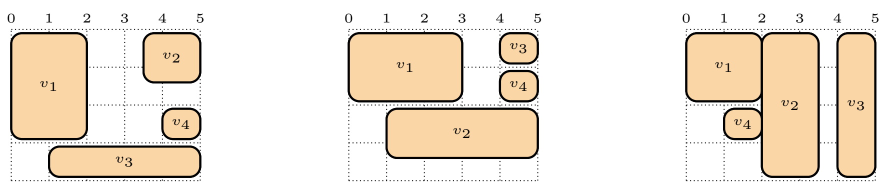
Given the total energy (second column) possible ranges for the rates, minimum start times and maximum end times (3rd to 5th columns), bounds consistency on the duration and demand variables yields the ranges shown in the 6th and 7th rows, respectively.

Therefore, a standard cumulative constraint will use the lower bounds and consider four tasks of durations and consumptions 2, $\frac{3}{2}$, 1 and 1, respectively. The classic `cumulative` constraint will not adjust the domains further, as shown in

Table 1: Populations, maximum rates, minimum start times and maximum end times of four evacuation tasks

	w_i	m_i	r_i	d_i	duration	height
v_1	6	3	0	3	$[2, 3]$	$[2, 3]$
v_2	6	4	1	5	$[\frac{3}{2}, 4]$	$[\frac{3}{2}, 4]$
v_3	4	4	1	5	$[1, 4]$	$[1, 4]$
v_4	4	4	1	5	$[1, 4]$	$[1, 4]$

Figure 1. The set of 3 solutions illustrated in this figure shows that every bound of every one of the eight variables (start times and height for each of the four evacuation tasks) is consistent[1].



(a) (b) (c)

Fig. 1: Some feasible schedules

4 Flow Global Constraint

The standard `cumulative` constraint is weak for this class of problems since it is unable to reason about total energy of each task. We can take advantage of the knowledge that the product (duration $\times$ rate) is a constant by using a more global constraint.

Given a set of tasks J with start time $s_i \in [r_i, \infty]$, completion time $e_i \in [0, d_i]$, height $h_i \in [0, m_i]$, constant energy w_i and a resource r of constant capacity q_r, `energetic_cumulative`$((s_i, e_i, h_i, w_i)_{i \in J}, q_r)$ enforces the relations:

$$\sum_{i \in J | s_i \leq t \leq e_i} h_i \leq q_r \quad \forall t \in \mathcal{H} \tag{10}$$

$$w_i = h_i(e_i - s_i) \quad \forall i \in J \tag{11}$$

[1] Tasks v_4 and v_3 are symmetric so we omit the bounds for v_4

In other words, `energetic_cumulative` is the conjunction of constraints (5) and (6) of Section 3.

The propagator works by building a flow network relaxation $f(\mathcal{D})$ of the problem, before it propagates, using the current domain $\mathcal{D}$.

Given current domain $\mathcal{D}$, let $T_V = \{lb(s_v), lb(e_v), ub(s_v), ub(e_v) \mid v \in V\}$ be the set of $O(n)$ minimum or maximum start and end times of the tasks of V and let t_i be the i-th largest element of T_V. We partition the time line into the set of consecutive intervals $I(V) = \{[t_i, t_{i+1}[| \; 1 \leq i < |T_V|\}$

Next, we create a flow $f(\mathcal{D})$ network as follows. We have source node S and a first layer of task nodes V, the flow from S to v is w_v for each v. We create a second layer of time interval nodes $I_i = [t_i, t_{i+1}[, 1 \leq i \leq n-1$. There is an edge from v to I_i if $lb(s_v) \leq t_i \wedge t_i + 1 \leq ub(e_v)$. The edge is bounded by $0..(t_{i+1} - t_i) \times ub(h_v)$. We create a final layer to a sink node E. There is an edge from each interval node I_i to E with capacity bounded by $(t_{i+1} - t_i) \times q$.

Theorem 1. *Any solution to* `energetic_cumulative`*($\bar{s}$, $\bar{e}$, $\bar{w}$, $\bar{h}$, q) given current domain $\mathcal{D}$, is extendible to a solution of the flow network $f(\mathcal{D})$.*

Proof. Given a solution for `energetic_cumulative`($\bar{s}$, $\bar{e}$, $\bar{w}$, $\bar{h}$, q) we extend it to a solution of the flow network $f(\mathcal{D})$ as follows. The flow from S to each node v is set to w_v. The flow from each node v to I_i is set to $(\min(e_v, t_{i+1}) - \max(s_v, t_i)) \times h_v$ The flow from each node I_i to E is given by the sum of the incoming flows to I_i. We show that these flows obey the bounds and flow balance equations. Examining each node v, the flow in is w_v and the flow out is $h_v \times (e_v - s_v)$. These must be equal by equation (6). Examining each node I_i the flow in is $\sum v \in \mathcal{E}((\min(e_v, t_{i+1}) - \max(s_v, t_i)) \times h_v$ but by equation (5) at no time is there more than q resource being used, hence this is no more than $(t_{i+1} - t_i) \times q$, the capacity of the outgoing arc. The flow balance at I_i holds by construction. □

Example 2. If we consider the circumstances explained in Example 1, the constraint `energetic_cumulative` generates the flow network shown in Figure 2.

While the standard propagate can determine nothing, the flow network is infeasible, so the `energetic_cumulative` propagator immediately fails.

5 Generating a Realistic Data Set

Catastrophic wildfire requiring large population evacuation are, thankfully, rare events. However, it means that obtaining useful data is difficult, and indeed this a key problem within the GEOSAFE project. A significant part of the project revolves around simulation tools such as EXODUS [6], however, even simulated data was hard to come by.

Therefore, we opted for taking advantage of the project environment to contribute to this effort by generating our own "realistic" dataset. On the one hand, this approach may introduce biases since we must use models to generate realistic road networks and simulate wildfires. On the other hand, we believe that it

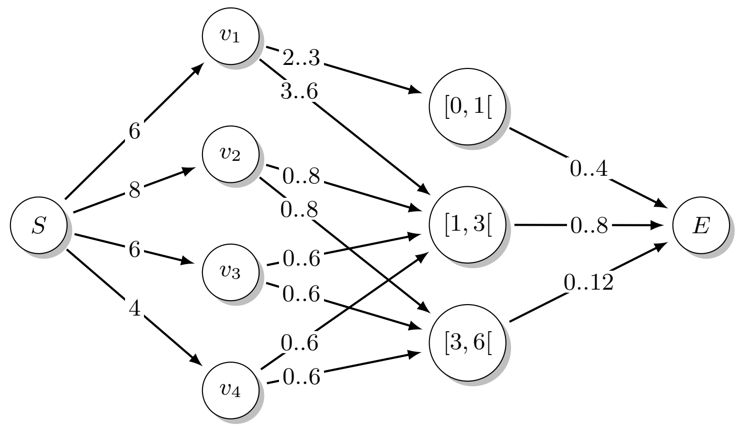
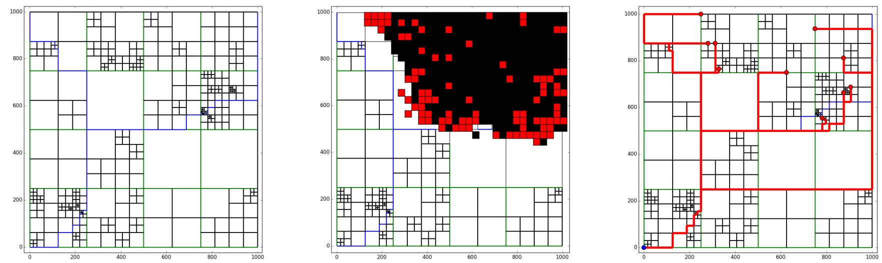


Fig. 2: Flow network relaxation of the `energetic_cumulative` constraint for the state given in Table 1.

will make it much more convenient for benchmarking algorithms in the future. As it turns out, the generated instance are very challenging even though relively modest in size, thus being interesting from an academic viewpoint as well.

5.1 Generation of road networks

The first step is to generate a graph standing for the road network. To this end, we used the *quadtree* model described in [3]. In a nutshell, this model starts with a single square formed by four nodes and four edges. At each iteration, a square is chosen and five nodes are added, one in the center of the square, and one on each edge connected by a perpendicular edge to the center node. A parameter r controls the *sprawl*, that is, the prefence for splitting larger squares ($r < 1$) or smaller square ($r > 1$). The graphs generated in this manner share a many features with real road networks: they are planar, embedded in an Euclidean plane, have similar density distributions, path lengths are within a constant factor of the Euclidean distance, and the number of turns is logarithmic with high probability. An example of random quadtree network is shown in Figure 3a. The colors on the edges correspond to road capacity. We first compute a minimum Steiner tree spanning three randomly chosen nodes in high density areas ("cities") and connect these cities to the nearest corner of the outer square. The corresponding set of edges are given the highest capacity and are coloured in blue in Figure 3a. A second set of edges, forming a grid are given an intermediate capacity, they are coloured in green.



(a) road network (b) simulated wildfire (c) evacuation plan

Fig. 3: An example of generated instance

5.2 Simulating wildfire

The second step consists in determining *safety due dates* for every edge of the evacuation tree, that is, a time after which the edge become unsafe. To this purpose we use a relatively simple fire propagation model. We chose to use a simple model based on two parameters: a constant *intensity* γ representing the type of fuel material as well as the temperature, and a *wind direction*. Indeed, the goal is not so much to accurately predict fire propagation, as it is to generate safety due dates consistent with a wildfire. Of course, should the authorities use this type of planning tools during a real event, then correctly predicting fire propagation would be among the most important factor.

The landa area is discretized into squares of fixed size (we use another parameter to control this size) which can be in three states: *untouched*, *burning* and *burned*. The fire starts as a single burning square, then at each iteration, any untouched square adjacent of a burning square catches fire with probability:

$$\gamma(\frac{\pi - A}{\pi})^2$$

Where γ stands for the intensity of the fire, and A is the angle between the wind an a vector going from the center of burning square to the center of the untouched square. Moreover, any burning square that did not propagate stop buring with probability γ^2. Figure 3b illustrate the state of the simulated wildfire, with burning squares in red and burned squares in black.

5.3 Generating evacuation plans

The third step consists in generating the actual evacuation plan, that is, an embedded tree connecting a set of evacuation nodes $\mathcal{E}$ to a safe node r. Here again, the goal is not to compute the best evacuation plans, however they must be representative of what would be actual plans.

We first randomly pick a predefined number of evacuation nodes among the nodes of the graph that are in the state *burned* or *burning* of the simulated fire. Then we use the convention that the safe zone is the furthest corner from the center of the fire. The evacuation tree is computed simply by using a shortest paths algorithm, however with respect to an arc labeling taking into account first the safety due date of the arc, and only then its length and its capacity.

At this point we have all the information we need to define a fire evacuation problem as defined in Section 2. However, we use Observation 1 to remove redundant arcs. For every evacuation path, we explore the arcs from the safe node r to an evacuation node v. For any section of the path on which nodes have single child, all the nodes of that section have the same set of descendant leaves in the tree. Therefore the same set of evacuees will go through these nodes with the same relative delay. Therefore, we conserve only the arc of minimum capacity in that section, and only if this capacity is strictly smaller than that of the previous section.

It follows that an instance with n evacuation tasks has at most $O(n)$ "jam" constraints. Moreover, the objective can be stated as the maximum of $O(n)$ expressions as shown in Section 2.

The tools we developed as well as the benchmarks instances we used in this paper can be accessed here: tbd.

6 Experimental Results

We generated 240 benchmark instances following the protocol described in Section 5. They are organized into three types of road networks: `Dense`, `Medium` and `Sparse` where the density refers to the number of intersections (respectively 400, 800 and 1200) in the land area. Notice that the graph has always 4 edges per node, so this corresponds to graph size. The impact on the instance is that generally, a larger graph allows for more choices for the shortest paths and therefore longer independent paths. For every type of road network, we generated 4 classes of instances, with respectively 10, 15, 20 and 25 evacuation nodes. Finally, for every class we simulated 20 random wildfires and the subsequents evacuation trees.

We used CPLEX to solve the flow formulation, which turned out to be extremely costly. Therefore we did not systematically called the propagator at each node. Instead we implemented a heuristice method to decide whether we should solve the flow using two criteria:

The first criterion we use is the total load $\Omega(r)$ of a resource r or capacity q_r on the tasks J:

$$\Omega(r) = \frac{\sum_{i \in J} w_i}{q_r(\max_{i \in J} e_i - \min_{i \in J} s_i)}$$

We only call the flow when $\max_{u \in \tilde{\mathcal{T}}} \Omega(u) \geq 0.95$, this value was empirically chosen. Moreover, observe that when $\Omega(r) > 1$ the constraint is trivially infeasible. Therefore we can return an early failure in this case.

The second criterion is the size of the remaining search space. Indeed solving the flow migh not be worthwhile even it is infeasible, if brute-force search would have been faster. Here we empirically chose 2^{50} as minimal search space, defined as the product of the ranges of all start time and heigh variables. That is, we solve the flow only when:

$$\sum_{i \in \mathcal{E}} \log |D(s_i)| + \log |D(h_i)| \geq 50$$

However, even with this approach, using the global flow constraint made CP Optimizer about one order of magnitude slower.

Table 2: Depth first search: upper bound and optimality ratio

		DFS		DFS^h		DFS$^+$		DFS^{h+}		DFS^f		DFS^{hf}	
		#s	ub	#s	ub	#s	ub	#s	ub	#s	ub	#s	ub
dense_10	(20)	1.00	33478	1.00	28862	1.00	22286	1.00	27836	1.00	23787	1.00	33700
dense_15	(20)	1.00	179099	1.00	179570	1.00	201867	1.00	199987	1.00	211582	1.00	212471
dense_20	(20)	1.00	365596	1.00	367030	1.00	406828	1.00	407116	1.00	490221	1.00	448120
dense_25	(20)	0.95	697661	1.00	663314	0.85	1093784	1.00	759437	0.85	1201755	1.00	881680
medium_10	(20)	1.00	49294	1.00	48159	1.00	48446	1.00	48446	1.00	49567	1.00	49567
medium_15	(20)	1.00	166737	1.00	156084	1.00	166498	1.00	156432	1.00	190724	1.00	195434
medium_20	(20)	0.95	339103	1.00	339370	1.00	376395	1.00	384381	1.00	555311	1.00	428312
medium_25	(20)	0.90	1291811	1.00	659269	0.80	1461162	1.00	722167	0.75	1292794	1.00	791500
sparse_10	(20)	1.00	4474	1.00	4759	1.00	4464	1.00	5951	1.00	4464	1.00	5951
sparse_15	(20)	1.00	131229	1.00	129276	1.00	135192	1.00	134420	1.00	147785	1.00	147071
sparse_20	(20)	0.90	293930	1.00	312834	1.00	320080	1.00	323449	1.00	439340	1.00	364677
sparse_25	(20)	0.95	598501	1.00	574534	0.80	593962	1.00	732975	0.80	685214	1.00	768972
avg	(240)	0.97	340545	1.00	292308	0.95	381291	1.00	329444	0.95	418918	1.00	366166

Table 3: Default search: upper bound and optimality ratio

		CPO		CPO^h		CPO$^+$		CPO^{h+}		CPO^f		CPO^{hf}	
		ub	opt	ub	opt	ub	opt	ub	opt	ub	opt	ub	opt
dense_10	(20)	23294	0.95	23567	0.80	23354	0.95	23681	0.85	23503	0.90	23807	0.80
dense_15	(20)	161100	0.70	161700	0.75	162856	0.60	162751	0.55	164524	0.45	165025	0.45
dense_20	(20)	311676	0.45	312501	0.45	312697	0.15	313114	0.15	317035	0.10	316524	0.00
dense_25	(20)	531016	0.00	529889	0.00	530749	0.00	530975	0.00	539076	0.00	564775	0.00
medium_10	(20)	48591	1.00	48591	0.80	48591	0.95	48591	0.75	48776	0.90	48979	0.70
medium_15	(20)	124921	0.70	125195	0.50	125559	0.60	125110	0.60	126011	0.40	127421	0.40
medium_20	(20)	276101	0.25	276094	0.25	277545	0.10	277485	0.10	279654	0.05	278840	0.10
medium_25	(20)	488282	0.10	488555	0.10	491794	0.00	491423	0.00	493616	0.00	505519	0.00
sparse_10	(20)	3747	1.00	3880	0.90	3747	1.00	3880	0.90	3946	0.85	3880	0.85
sparse_15	(20)	120173	0.65	120151	0.65	120162	0.60	119803	0.55	121083	0.55	120576	0.55
sparse_20	(20)	235771	0.35	253807	0.35	237503	0.15	236920	0.20	238300	0.20	240393	0.15
sparse_25	(20)	437618	0.00	437219	0.05	439699	0.00	438160	0.00	441659	0.00	448979	0.00
avg	(240)	230191	0.51	231762	0.47	231188	0.42	230991	0.39	233098	0.37	237060	0.33

7 Conclusion

PJS: We can make use of dual network flow to propagate bounds on start and end times and rates (Algorithm from nnote.tex). PJS: We could make the flow network construction incremental.

References

1. Hal E. Anderson. Aids to determining fuel models for estimating fire behavior. Technical Report 122, United States Department of Agriculture- Forest Service, Intermountain Forest and Range Experiment Station Ogden, UT 84401, April 1982.
2. Vedat Bayram. Optimization models for large scale network evacuation planning and management: A literature review. *Surveys in Operations Research and Management Science*, 2016. DOI: 10.1016/j.sorms.2016.11.001.
3. David Eisenstat. Random road networks: the quadtree model. *CoRR*, abs/1008.4916, 2010.
4. Caroline Even, Victor Pillac, and Pascal Van Hentenryck. Convergent plans for large-scale evacuations. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 1121–1127, 2015.
5. Caroline Even, Andreas Schutt, and Pascal Van Hentenryck. A constraint programming approach for non-preemptive evacuation scheduling. In *Principles and Practice of Constraint Programming - 21st International Conference, CP 2015, Cork, Ireland, August 31 - September 4, 2015, Proceedings*, pages 574–591, 2015.
6. Edward R. Galea, Mathew Owen, and Peter J. Lawrence. The EXODUS Model. *Fire Engineers Journal*, pages 26–30, 1996.
7. Geo-safe - geospatial based environment for optimisation systems addressing fire emergencies, MSCA-RISE-2015 - Marie Skodowska-Curie Research and Innovation Staff Exchange (RISE) European projet – id 691161. http://fseg.gre.ac.uk/fire/geo-safe.html. Accessed: April 21, 2018.
8. Kanal Kumar, Julia Romanski, and Pascal Van Hentenryck. Optimizing infrastructure enhancements for evacuation planning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 3864–3870, 2016.
9. Victor Pillac, Manuel Cebrián, and Pascal Van Hentenryck. A column-generation approach for joint mobilization and evacuation planning. *Constraints*, 20(3):285–303, 2015.
10. Victor Pillac, Caroline Even, and Pascal Van Hentenryck. A conflict-based path-generation heuristic for evacuation planning. *Transportation research part B*, (83):136–150, 2016.
11. Richard C. Rothermel. A mathematical model for fire spread predictions in wildland fuels. Technical Report 115, USDA For. Serv., lntermt. For. and Range Exp. Stn., Ogden, Utah, USA, 1972.
12. Shahrooz Shahparvari. *Enhancing Emergency Response in Short-notice Bushfire Evacuation*. PhD thesis, RMIT University, 2016.
13. Shahrooz Shahparvari, Prem Chhetri, Babak Abbasi, and Ahmad Abareshi. Enhancing emergency evacuation response of late evacuees: Revisiting the case of australian black saturday bushfire. *Transportation Research Part E: Logistics and Transportation Review*, 93:148 – 176, 2016.

14. Alexander Stepanov and James MacGregor Smith. Modeling wildfire propagation with delaunay triangulation and shortest path algorithms. *European Journal of Operational Research*, 218(3):775 – 788, 2012.
15. Anand Veeraswamy, Edwin R Galea, Lazaros Filippidis, Peter J Lawrence, and Robert J Gazzard. The simulation of urban-scale evacuation scenarios: Swinley forest fire. In *Proceedings 6th Int Symp Human Behaviour in Fire*, pages 221–232, 2015.
16. David R. Weise and Gregory S. Biging. A qualitative comparison of fire spread models incorporating wind and slope effects. *Forest Science*, 43(2):170–180, 1997.