FUNDAMENTAL RESEARCH

forest management

Generating Stochastic Ellipsoidal Forest and Wildland Fire Scar Scenarios for Strategic Forest Management Planning under Uncertainty

AQ: A **Claudio A. Kuhlmann, David L. Martell, Roger J-B Wets, and David L. Woodruff**

Explicit consideration of the possibility of wildfires enhances strategic planning for landscape management; however, incorporating the stochastic processes that govern fire occurrence and spread in optimization models that are used to inform landscape management presents new challenges. In this article, we describe a method of generating spatially explicit fire scenarios in the form of elliptical fire scars with associated probabilities. Computational experiments demonstrate that good scenarios can be achieved using currently available computer technologies. We also describe how more complex fire ignition and growth process models can be used to generate more realistic fire pattern scenarios and provide insight into the quality and effort required for various approximations. We conclude by discussing how our methodology could be used to perform some of the tasks for which fire managers use traditional burn probability models.

Keywords: stochastic fire scenarios, forest management, landscape management, elliptical fire scars

Forests generate many social, economic, and ecological benefits but they are complex dynamic ecosystems that are often influenced by stochastic fire, insect, disease, and storm events that forest ecologists and forest managers often refer to as disturbances.[1] Forest managers are responsible for the development and implementation of strategic, tactical, and operational plans that specify, for example, when and where protected areas are to be established, what areas are to be used for recreation purposes, when and where forest access roads are to be established, when specific forest stands will be harvested for industrial purposes, and how the resulting cut-overs will be regenerated, tended, and protected until the next time they are scheduled to be harvested. Strategic planning horizons in the boreal forest region of Canada span 100 years or more during which tactical plans with planning horizons of several years and daily, weekly, and monthly operational plans are also developed and implemented. Uncertainty concerning natural disturbance processes and many other factors such as, for example, the future cost of harvesting and processing forest resources, the prices at which forest products might be sold, and societal preferences concerning how forests can and should be managed all complicate forest management planning. Recent years have witnessed growing interest in forest management planning under uncertainty. Hanewinkel et al. (2011) provide a comprehensive review of risk management in forestry, whereas Thompson and Calkin (2011) focus on dealing with risk in wildland fire management. In this article, we describe the development and use of a novel approach for incorporating uncertainty concerning fire in forest management planning.

Fires can ignite and spread almost anywhere across forested landscapes so the state space of possible futures that forest managers must consider (i.e., the fire perimeters or burn scars that might appear on the landscape) is continuous, which can be a challenge for forest managers who seek to address such uncertainty when they develop spatially explicit forest management plans. Deterministic linear programming models have been used extensively to support strategic management planning in many jurisdictions since the early 1970s (see, e.g., Navon 1971), and one of the accepted ways of incorporating uncertainty in planning processes in which linear programming models are used is to develop stochastic linear programming (SLP) models that are structured to account for uncertain futures (see, e.g., King and Wallace 2010).

One SLP approach calls for the generation of a discrete set of plausible scenarios that describe important aspects of what might

occur in the future, estimation of the probabilities that those scenarios might occur, and the incorporation of such information in SLP models. Gassmann (1989) and Boychuk and Martell (1996) developed multistage aspatial SLP models that can be used to develop aspatial strategies for timber harvest scheduling under uncertainty due to fire. In this article, we develop a methodology for using models of continuous state space stochastic fire ignition and spread processes to generate a set of discrete fire scar pattern scenarios and estimate the probabilities that those scenarios will occur in the future. From the perspective of stochastic programming, this is novel because it is the first work to form scenarios that are shapes on a plane. Both the discretization and the probability estimation are contributions of this work.

The rest of the article proceeds as follows. We begin by placing our work in the context of the extant literature and providing an overview of the issues. Next we lay out the method we propose for scenario generation, which creates a need to determine distances between elliptical fire scars that we address. Our computational experiments are then described and the article ends with our discussion and conclusions section in which we outline directions for further research and application.

Forest Management Planning under Uncertainty

Although forest harvesting and regeneration in the boreal forest region of the province of Ontario, Canada, take place at the level of stands that are typically on the order of 25 ha or less, our focus is on strategic management at the landscape or forest management unit level, which might extend across 100,000 ha or more. When forest managers develop forest management plans for flammable forest landscapes, they must determine to what extent and how they should incorporate potential but uncertain fire losses in their plans.

Martell (1980), Routledge (1980), and Reed (1984) extended the traditional deterministic stand-level Faustmann optimal stand rotation model to account for fire losses and found, not surprisingly, that the optimal planned rotation interval should decrease as the probability that a stand will burn increases. Unfortunately, their results cannot be applied to the management of large landscapes because they do not account for many factors including, for example, the facts that the burning of a stand may affect the probability that nearby stands will burn and that unconstrained optimal stand-level decisions are unlikely to provide the relatively stable harvest flows required by mills.

Forest managers began using aspatial deterministic linear programming (LP) models to develop strategic forest management planning models in the early 1970s (see, e.g., Navon 1971). Specific stands were aggregated into groups or strata that were similar with respect to age, species, growth rates, and other stand attributes, and LP models were used to determine the annual allowable cut: how much area should be harvested from each stratum during each period to maximize some objective function (e.g., present net worth) subject to constraints (e.g., variability in period to period harvest flows) over a 100-year planning horizon. Those LP models did not account for fire, and they were aspatial inasmuch as they did not specify which stands should be harvested during each period.

Reed and Errico (1986) developed a "mean value" strategic forest management planning model in which they addressed uncertain fire processes by assuming that some "known" average fraction of the forest burns during each time period. They conjectured that for forests that did not have large burn rates, the solutions to their mean value model would be close to the solutions to the corresponding

stochastic models, were they available. Gassmann (1989) formulated a smaller version of the Reed and Errico (1986) model as a multistage stochastic programming model and dealt with the desire to mitigate changes in harvest flow by including a harvest flow reduction penalty in his objective function. Boychuk and Martell (1996) built on Gassmann's approach and developed a stochastic programming model which they used to investigate alternative strategies for managing a hypothetical flammable forest that was representative of the boreal forest region of the province of Ontario. They found that Reed and Errico's (1986) conjecture was valid for fire activity levels typically observed in forest management units in the boreal forest region of the province of Ontario.

Recent years have witnessed tremendous growth in interest in spatially explicit models driven in part by widespread recognition that in many countries (e.g., parts of Australia, Canada, and the United States) forest vegetation and fuel buildups have made it necessary to determine when and where to treat fuels to mitigate the detrimental impacts of fire. Some authors have incorporated simple fire ignition and spread process models in spatially explicit integer and dynamic programming models (e.g., Wei et al. 2008, Konoshima et al. 2010), but most efforts have been directed to using simulation methods to develop what are referred to as burn probability (BP) models that can be used to evaluate fuel management strategies.

Most BP models use simulation methods to provide estimates of the probability that specified points on a flammable landscape will burn each year. The landscape is usually partitioned into a large number of small cells or pixels and fire ignition and growth processes are simulated for many (typically thousands) of years or fire seasons. The simulation model tracks how often each pixel burns and the number of years that a pixel burns divided by the number of years simulated is an estimate of the probability that pixel will burn each year. Many BP models have been developed for many different landscapes (see for example, Miller et al. 2008, Ager et al. 2012, Scott et al. 2012) and range in complexity from those that are based on very simple fire ignition and spread process models to those that are based on more complex fire ignition, fire suppression, and spread on landscapes for which detailed descriptions of vegetation or fuel and topographical features are provided.

BP models can be used to develop very powerful decision support systems that can be used to enhance the management of flammable landscapes, and they can be produced relatively easily once one has compiled a digital description of the fuel and topography of the landscape and coupled it with weather data and software implementations of fire ignition, suppression, and growth process models. They do, however, have some limitations.

Many BP models are based on implicit assumptions that all the simulated scenarios used to generate the estimated probabilities are equally likely, which is unlikely. They produce point estimates of the probability that any pixel on a landscape will burn during a designated time period (e.g., a year or fire season), but they do not provide confidence limits on those estimates. Furthermore, because fire ignition, suppression, and spread are inherently spatial processes, burn probabilities are actually spatially correlated.

Suppose one has a set of cells I and a second set of cells J on a landscape. Because the cell-level burn probabilities are not independent, one could not easily use them to estimate the probability that all of the cells in set I and none of the cells in set J would burn during any particular period. Thompson et al. (2013) address that concern by focusing on polygons that contain what they describe as highly valued resources and assets (HVRA) that can be damaged by fire,

partitioning those polygons into cells, and then determining how often simulated fire scars overlap the cells in those polygons. They are then able to use their results to generate attributes of the burn probability distributions for each of the HVRA polygons such as the proportion of times some specified fraction of the cells within that polygon are burned.

In this article, we develop a methodology that can be used to produce what we describe as stochastic fire scar patterns for which one can estimate the probability of occurrence that can be used to answer such questions. Our approach is to incorporate the type of fire ignition, suppression, and growth models that others have used to produce traditional BP maps in a simulation model that we use to generate a discrete set of fire scar patterns and estimate the probability that each of those patterns will occur each year. Although most BP models are based on gridded landscapes (see Miller et al. 2008), we model fire ignition and spread in continuous time and space.

We begin by describing how we used a fire ignition and an elliptical growth model to generate spatially explicit fire scenarios in the form of elliptical fire scars with associated probabilities for a hypothetical forest that is representative of the boreal forest region of Canada. Our computational experiments demonstrate that good scenarios can be achieved using currently available computer technologies.

Simulating Fire Ignition and Growth

To generate fire scar patterns we need to simulate the ignition, suppression, and growth of fires. Because our primary objective is to develop a methodology for generating spatially explicit stochastic fire scar patterns, we chose to use a very simple fire model that can readily be replaced by a more realistic model for more complex or real flammable landscapes.

The number of lightning-caused fires that are ignited, detected, and reported and burn a significant portion of a forest management unit each year is the result of a uncertain number of lightning storms that pass over the area, the number of cloud-to-ground strokes produced by each storm, the moisture content and other attributes of the forest vegetation or fuel those lightning strokes strike, the subsequent weather, and the effectiveness of the detection and suppression systems. Because we are primarily interested in generating spatially explicit stochastic fire scar scenarios and we are not aware of any well-validated models of a fire management system that includes all of those processes and their interaction, we used a very simple fire ignition and growth model to generate our fire scars.

Most of the forest fires that occur in the province of Ontario are controlled by the initial attack force and burn little or no area, and most forest management units in Ontario experience at most one large (e.g., $\geq$200 ha) fire each year. Given the need to start with very simple stochastic fire scar scenarios, we decided to restrict our attention to lightning-caused fire scar patterns with at most one large fire per year.

We did so by assuming that our forest will be struck by at most one cloud-to-ground lightning stroke each year and that the probability of a lightning strike igniting a fire that will survive the rainfall associated with the storm cloud depends on many factors including the type of vegetation or fuel that it strikes and being reported (arrives and demands the attention of the fire suppression system) is P_{ig}. We further assume that the probability the fire would escape the initial attack and grow across the landscape to become a large fire is P_{esc}.

Our model forest is 10×10 km or 10,000 ha in size, and the final size of escaped fires is assumed to be exponentially distributed with an expected value of 200 ha. Fires grow in complex shapes that are determined by the fuel and topography in which they are burning, the weather, particularly the wind speed and direction, and suppression efforts. Many of the large fires that burn across the boreal forest region of Ontario are somewhat elliptical in shape, which is consistent with the simple elliptical fire spread model first described by Van Wagner (1969). Because we are assuming that our forest is flat and has a homogeneous fuel composition, we also assumed that all of our fires burn in the shape of an ellipse with a width/breadth ratio of 2.0 We assumed that the direction in which the major axis of the ellipse is oriented is normally distributed with a southwest to northeast direction, which is representative of large fires in the boreal forest region of Ontario that are most often driven by relatively dry southwest winds.

Most of the large fires that burn in the province of Ontario exhibit significant temporal variation in their growth rates. They grow rapidly on days when weather conditions favor fire growth until they experience a fire-ending event (e.g., very significant rainfall and or successful suppression action), but they grow very slowly at night. Many experience fast growth rates for a very small number (in same cases only one) of days that are interspersed in sequences of low-hazard days during which they exhibit little or no significant growth. Empirical distributions of fire sizes in the boreal forest are very tail heavy, and we assumed, for simplicity, that the final size of our escaped fires had an exponential distribution.

The net result of our simplifying assumptions is that in our simulated forest, each year at most one large fire burns in the shape of an ellipse, the origin of which is uniformly distributed across our hypothetical landscape and the orientation of which is stochastic to produce a set of *fire scenarios* that can be incorporated in scenario-based stochastic optimization models (Valsta 1992, Gassmann and Ireland 1995). Because the scenario for no fire is easy to characterize, we proceed to describe the process of generating single-fire scenarios. The algorithm for generating the fire scar scenarios can be summed up as follows

1. Find S_1 fire scars, by doing S_1 simulations to form the set of fire cell sets (i.e., fire scars) called ε.

2. Run S_2 simulations and assign each fire to its nearest fire scar in ε.

3. The probability associated with each member of ε is the number of assignments divided by S_2 times the probability of a fire.

The present analysis assumes only one significant fire per season or time period under consideration. In step 3, the probability calculation allows for the possibility of no fire by unconditioning the probabilities using the probability of a fire.

Step 1 discretizes the scenario space (the space of possible fires) to produce a set of representative fire scars ε. In the Representative Patterns subsection below, we describe experiments to determine approximately how many are needed to effectively discretize the space to generate discrete stochastic fire scenarios for the forest used in our experiments. Step 2 creates a need to compute distances between fire scars, which are elliptical sets of points in two dimensional space. This leads naturally to calculating the *classic Pompeiu-Hausdorff distance* between ellipses) (e.g., see Rockafellar and Wets

1998, §4.C). It measures how much a set A needs to be "fattened up" to cover the other set B and how much B would need to "grow" to cover A, taking the smaller of these two values

AQ: B

$$d_{PH}(A, B) = \min \eta \quad \text{such that} \quad \max_{a \in A} d(a, B) \leq \eta$$
$$\text{and} \max_{b \in B} d(b, A) \leq \eta$$

where $d(c, C)$ is the (minimal) distance between a point c and a set C.

Using the fire simulator, we obtain the parameters that define each ellipse: the major axis *(a)*, the minor axis (b), and rotation with respect to the x-axis (ϕ). The distance metric is the key to the algorithm for fire scar scenario generation that we just described so the next section is devoted to a study of methods for computing it.

Directed Pompeiu-Hausdorff Distance between Ellipses

In the case of simulated fire scars, the two sets of interest are ellipses, so we can take advantage of some special properties. The distances between two ellipses E_1 and E_2 is well parametrized in the x-y-axis by their semimajor axis (a_e), semiminor axis (b_e), center (xc_e, yc_e), and rotation angle (ϕ_e). With this, the *directed Pompeiu-Hausdorff distance* from E_2 to E_1 can be written as follows

$$\max_{(x_2, y_2) \in E_2} \min_{(x_1, y_1) \in E_1} \|(x_1, y_1), (x_2, y_2)\|_2 \tag{1}$$

Remember that because this is a directed distance, to obtain the Pompeiu-Hausdorff distance between E_1 and E_2 (which is not directional) one needs to solve this problem twice, from E_2 to E_1 and vice versa.

The solution to this problem can be obtained through different methodologies. Using the explicit representation of an ellipse, one can make use of nonlinear optimization tools. It is also possible to use several points to discretize the ellipses, allowing for the use of geometric properties of polygons.

Optimization Methods

To solve problem 1 by applying continuous optimization methods, it is helpful to manipulate ellipses E_1 and E_2 in a way that allows for a simpler representation of the problem. First, we rotate and translate the original axes to have E_1 centered and E_2 maintaining its original orientation with respect to E_1. To accomplish this, E_1 can be redefined using its canonical representation

$$\bar{E}_1 = \left\{ (\bar{x}_1, \bar{y}_1) \middle| \left(\frac{\bar{x}_1}{a_1}\right)^2 + \left(\frac{\bar{y}_1}{b_1}\right)^2 \leq 1 \right\} \tag{2}$$

$$\Rightarrow \bar{E}_1 = \left\{ (\bar{x}_1, \bar{y}_1) \middle| \left(\frac{\bar{x}_1}{a_1} \quad \frac{\bar{y}_1}{b_1} \quad 1\right) \in \mathcal{L}^3 \right\} \tag{3}$$

where

$$\mathcal{L}^k = \left\{ x \in \mathfrak{R}^k : x_k \geq \sqrt{\sum_{j=1}^{k-1} x_j^2} \right\}$$

is the k-dimensional Lorentz or second-order cone (Ben-Tal et al. 2009).

To define E_2 in the new axes it is useful to use the trigonometric parametrization of its border

$$E_2 = \Bigg\{ (x_2, y_2) \Big|$$

$$x_2(t) = xc_2 + a_2 \cdot \cos\phi_2 \cdot \cos t - b_2 \cdot \sin\phi_2 \cdot \sin t$$
$$y_2(t) = yc_2 + a_2 \cdot \sin\phi_2 \cdot \cos t + b_2 \cdot \cos\phi_2 \cdot \sin t$$

$$\forall t \in (0, 2\pi) \Bigg\} \tag{4}$$

It is also known (Eberly 2003) that the rotation and translation of an ellipse to a new origin (xc_1, yc_1) is given by

$$\begin{pmatrix} \bar{x}_2(t) \\ \bar{y}_2(t) \end{pmatrix} = \mathbf{R} \cdot \begin{pmatrix} x_2(t) - xc_1 \\ y_2(t) - yc_1 \end{pmatrix} \tag{5}$$

For this application,

$$\mathbf{R} = \begin{pmatrix} \cos - \phi_1 & -\sin - \phi_1 \\ \sin - \phi_1 & \cos - \phi_1 \end{pmatrix} = \begin{pmatrix} \cos\phi_1 & \sin\phi_1 \\ -\sin\phi_1 & \cos\phi_1 \end{pmatrix} \tag{6}$$

E_2 can then be described in the rotated and translated axes through the following equations

$$\bar{E}_2 = \Bigg\{ (\bar{x}_2, \bar{y}_2) \Big|$$

$$\bar{x}_2(t) = (x_2(t) - xc_1) \cdot \cos\phi_1 + (y_2(t) - yc_1) \cdot \sin\phi_1$$
$$\bar{y}_2(t) = -(x_2(t) - xc_1) \cdot \sin\phi_1 + (y_2(t) - yc_1) \cdot \cos\phi_1$$

$$\forall t \in (0, 2\pi) \Bigg\} \tag{7}$$

Rewriting problem 1 for ellipses $\bar{E}_1$ and $\bar{E}_2$ the following is obtained

$$\max_{(\bar{x}_2(t), \bar{y}_2(t)) \in \bar{E}_2} \min_{(\bar{x}_1, \bar{y}_1)} \|(\bar{x}_1, \bar{y}_1), (\bar{x}_2(t), \bar{y}_2(t))\|_2 \tag{8}$$

s.t.

$$\left(\frac{\bar{x}_1}{a_1} \quad \frac{\bar{y}_1}{b_1} \quad 1\right) \in \mathcal{L}^3$$

Three ways of solving this problem using optimization methods are proposed.

Benchmark Optimization Model

A first approach is to discretize $\bar{E}_2$ in N points and solve the minimization problem for each one of these. In this case, for any given point $(p, q) \in \bar{E}_2$, problem 9 needs to be solved.

$$\min_{(x, y)} \sqrt{(p - x)^2 + (q - y)^2} \tag{9}$$

s.t.

$$\left(\frac{x}{a_1}\right)^2 + \left(\frac{y}{b_1}\right)^2 \leq 1$$

$$x \in [-a_1, a_1]$$

$$y \in [-b_1, b_1]$$

The directed Pompeiu-Hausdorff distance is obtained by taking the maximum out of the N solutions obtained.
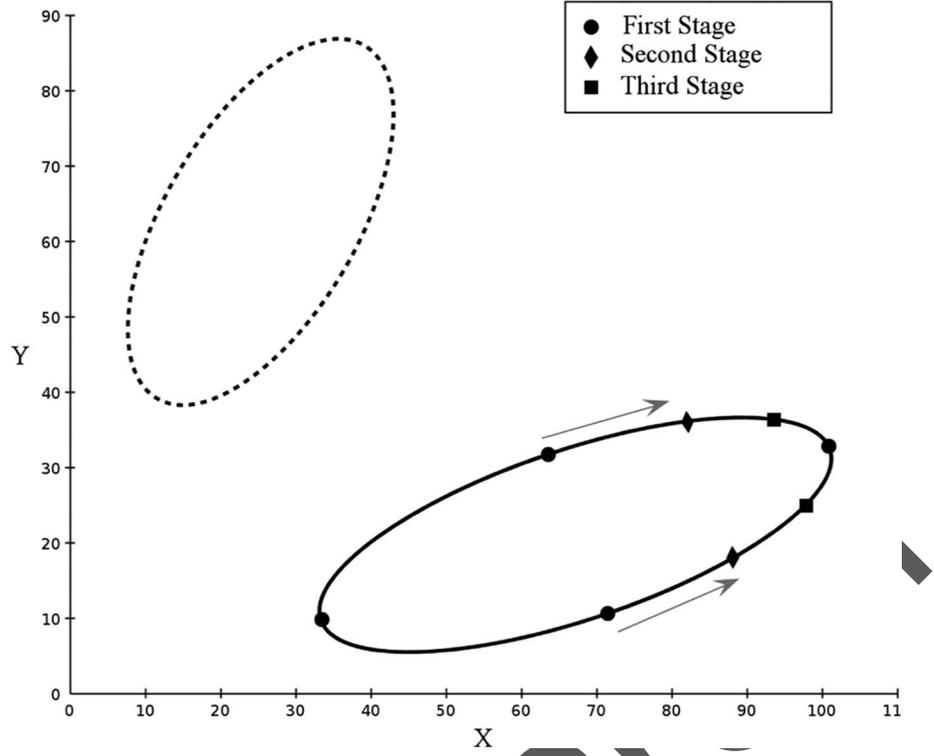
Figure 1. Binary search approach (shown without translation and rotation). The lower ellipse is $\bar{E}_2$.

This model is not the most practical one because it implies solving N optimization problems for each Pompeiu-Hausdorff distance calculation. However, it gives a very reliable answer and has a very straightforward implementation. For this reason, this method will be referenced as the *benchmark* method. Bear in mind that not all N problems need to be solved through optimization. For instance, if the point (p, q) lies inside $\bar{E}_1$, i.e., $(p/a_1)^2 + (q/b_1)^2 \leq 1$, then its distance to ellipse $\bar{E}_1$ is equal to 0.

Binary Search Benchmark Optimization Method

Exploiting the regularity of the figures in hand, it is possible to reduce the number of optimization problems to be solved. Based on the work described in Edelsbrunner (1985), we developed a heuristic method.

The method begins by calculating the distance between four equally spaced points of $\bar{E}_2$ and $\bar{E}_1$, after which the farthest point to $\bar{E}_1$ is picked. The method is based on this first approximation, as it then searches around this first point for the optimal solution. On each iteration, half of the remaining portion of the ellipse is dropped from the analysis, keeping the portion that is around the farthest point found up to that stage (Figure 1). With this heuristic, the number of optimization problems to be solved is reduced dramatically because it is not necessary to visit all the points of each ellipse. Consequently, the processing time is also reduced.

Conic Duality Optimization Model

A second possibility is to rewrite the minimization problem present in 8 as a maximization using conic duality (Ben-Tal et al. 2009, p. 415). This allows expression of the problem as a maximization-maximization instead of a maximization-minimization optimization problem. This method, referred to here as the *conic duality* or c-d method, is not so intuitive and requires some manipulation to derive its final representation. Details of this method can be found in the Appendix.

Contrary to what is desirable, this nonlinear optimization problem is not always concave. Its concavity depends on the ellipses to which the problem is applied (Figure 2). This is troublesome with nonlinear solvers such as KNITRO (Waltz and Nocedal 2003) and IPOPT (Wächter and Biegler 2006), which will sometimes end at a local optimum regardless of the existence of a better global optimum. To avoid this problem different initial values for the optimization variables α and β must be given, which transforms the method into a heuristic.

Bear in mind that every point of ellipse E_2 should be considered, both in its border and interior; however, only the border is relevant because that is where the optimization problem will find its solution. This approach is applied for the resolution of the benchmark as well as for the conic duality methods, in which only the border of the ellipse is considered in the discretization and parametrization, respectively.

Discrete Geometric Methods

It is also possible to solve problem 1 by discretizing E_1 and E_2 into n_1 and n_2 points, respectively, which can be thought of as converting them into two convex polygons P_1 and P_2. Having the two polygons in hand, one can apply one of the methods described below.

Exhaustive Search

In this method, we calculate the directed distance by computing for every vertex of P_2 the Euclidean distance between it and every vertex of P_1 to find the closest one. The maximum among these n_2 distances will be considered as the directed distance from P_2 to P_1. Following this procedure, the directed distance from E_1 to E_2 is

	a	b	(xc, yc)	$\phi[rad]$
E_1	100	60	(150,160)	2.019
E_2	200	75	(0,0)	0

(A)

	a	b	(xc, yc)	$\phi[rad]$
E_1	100	60	(0,50)	2.018
E_2	200	75	(0,0)	0

(B)

	a	b	(xc, yc)	$\phi[rad]$
E_1	0.738	101.63	(45.7,-22.8)	1.53
E_2	500.53	245.43	(0,0)	0

(C)
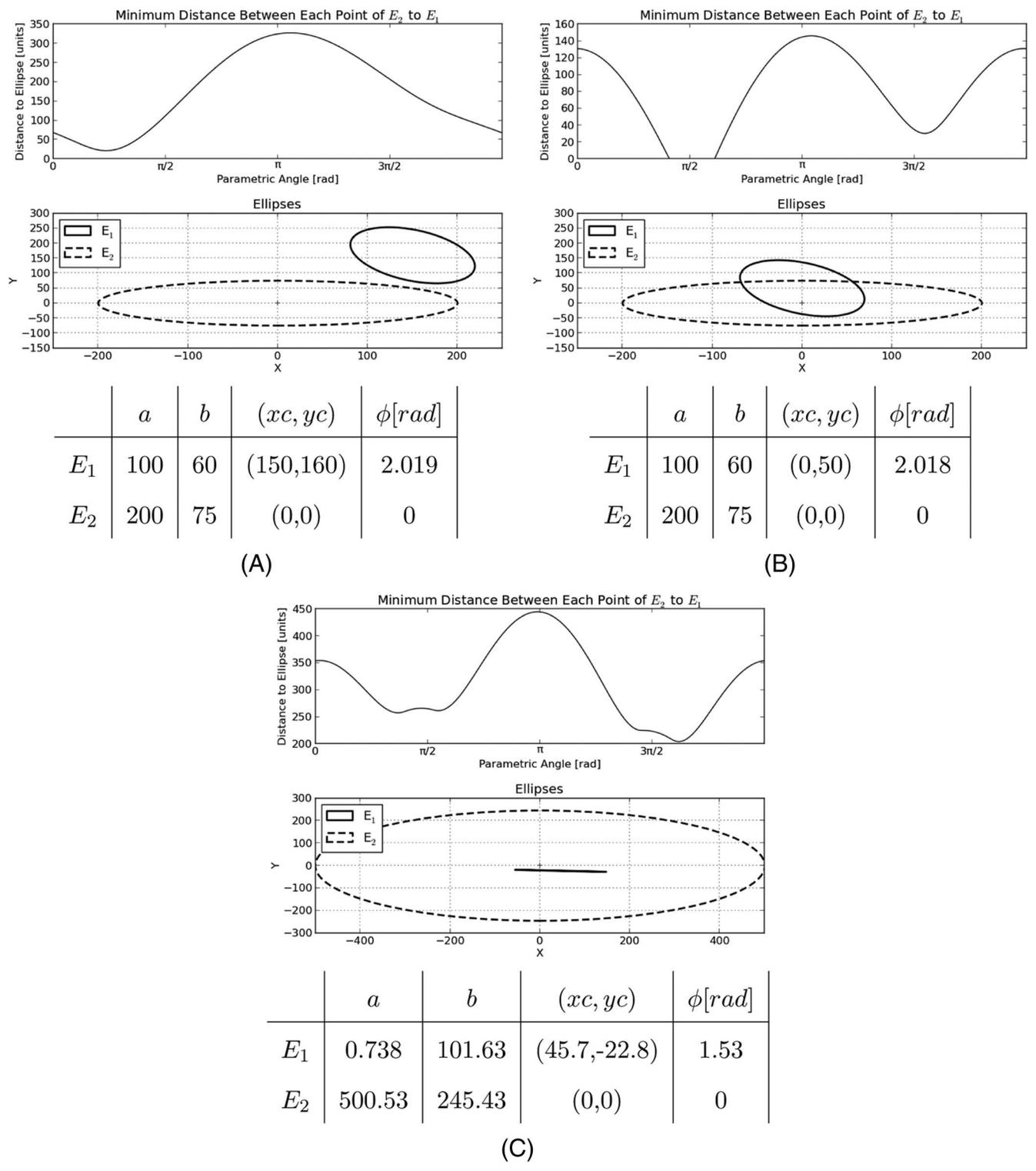
Figure 2. Examples of the directed Pompeiu-Hausdorff distance between different ellipses.

obtained. The Pompeiu-Hausdorff distance will then be the maximum between these two. In other words, a discrete Pompeiu-Hausdorff distance is computed in $\mathbb{O}(n_1 \cdot n_2)$ time (Figure 3 [the implicit polygons are not shown]). This is a trivial approach that most certainly could be improved.

It is important to note that this is not an exact method for calculating the Pompeiu-Hausdorff distance between convex polygons. It is quite possible that the closest point from a vertex of either polygon lies not on a vertex but on an edge of the other, something that is not considered by this approximation.

Atallah Algorithm

It is also possible to follow a more intelligent methodology to obtain the directed distance between P_1 and P_2, taking advantage of the properties shown by convex polygons. An algorithm that could compute the directed distance in $\mathbb{O}(n_1 \cdot n_2)$ time was first developed
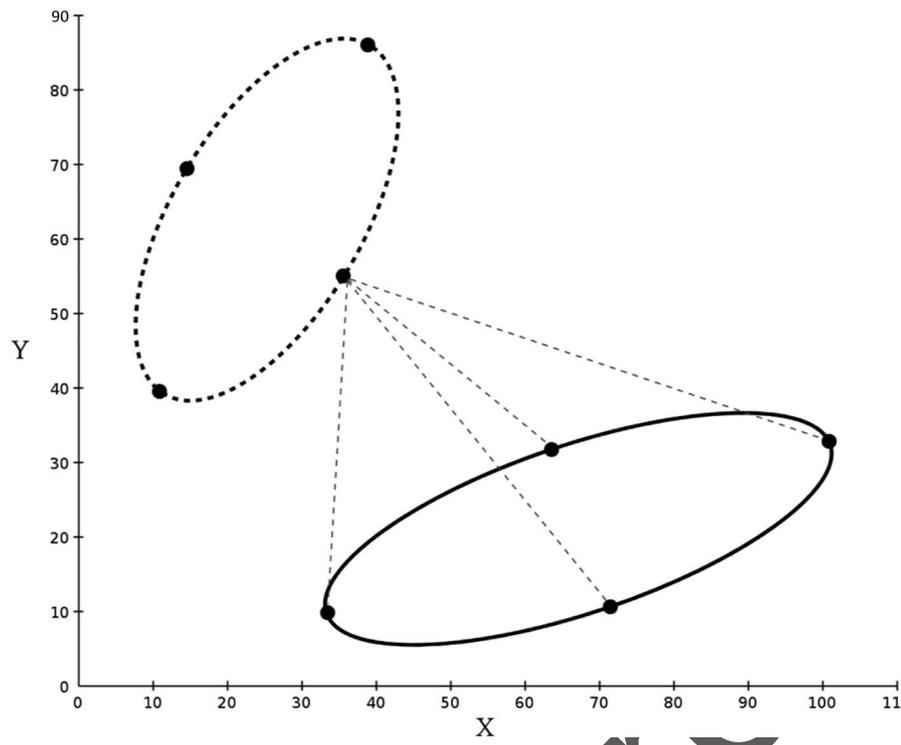
Figure 3. Ellipse approximation example for $n_1 = n_2 = 4$.

Table 1. Execution times for optimization methods with forest fire ellipses.

Method	N/starting points	Mean	SD
		. . . . . .(s)	
Benchmark	4	0.511	0.064
	50	9.150	1.488
	100	22.970	5.501
	400	207.038	82.986
Binary search benchmark	50	2.049	0.233
	100	2.790	0.580
	400	3.305	0.305
Conic duality	1	0.222	0.132
	4	0.991	0.560
	8	2.036	1.046

Table 2. Relative errors for optimization methods with forest fire ellipses.

Method	N/starting points	Mean	SD
		. . . . . .(%)	
Benchmark	4	1.179	1.475
	50	0.012	0.017
	100	0.003	0.005
	400		
Binary search benchmark	50	0.021	0.378
	100	0.003	0.005
	400	0	0
Conic duality	1	0.666	4.742
	4	−0.00021	0.0003
	8	−0.00021	0.0003

in Atallah (1983). However, it only considered disjoint convex polygons, which is unsuitable for the application in hand.

To explore this possibility, we implemented the algorithm developed in Atallah et al. (1991) which builds upon the previous work. By computing the possible intersections between the two polygons, which can also be made in $O(n_1 \cdot n_2)$ time (O'Rourke et al. 1982), the algorithm is able to calculate the directed distance for nondisjoint convex polygons, maintaining the complexity of the former algorithm.

For the implementation of both methods, we will consider an equal number of discrete points for both polygons (i.e., $n_1 = n_2 = N$).

Computational Experiments
Comparison of Methods

To assess the quantitative differences between the methods, we computed the Pompeiu-Hausdorff distance between 2,000 pairs of ellipses, divided into 10 sets, each one starting from a different seed

to generate the necessary random numbers. These experiments were conducted on an x86-64 machine with 2100-MHz CPUs and 126 GB of RAM. The benchmark method using 400 points was chosen to be the base model to which other models are compared to assess their accuracy. Results obtained using IPOPT solver (Wächter and Biegler 2006) are presented in the following sections.

Optimization Methods

Three optimization models were described: benchmark, benchmark with binary search, and conic duality. Table 1 shows that these methods vary considerably in their times of execution, depending on the number of discrete points N or the number of starting points, as it corresponds. Naturally, as more points or problems are considered, the computational time increases.

Table 2 shows the statistics for the relative error between every method and the chosen base method calculated as

$$\frac{\delta_{P-H}^{benchmark400} - \delta_{P-H}}{\delta_{P-H}^{benchmark400}} \qquad (10)$$

Table 3. Execution times for discrete geometric methods with forest fire ellipses.

Method	n	Mean	SD
		. . . .(s)	
Exhaustive search	4	0.0003	$9.161 \cdot 10^{-5}$
	50	0.011	0.003
	100	0.039	0.010
	400	0.563	0.103
	1,000	3.373	0.542
Atallah algorithm	4	0.005	0.001
	50	0.057	0.020
	100	0.120	0.055
	400	0.571	0.611
	1,000	1.909	3.473

Table 4. Relative errors for discrete geometric methods with forest fire ellipses.

Method	n	Mean	SD
		. . . .(%)	
Exhaustive search	4	−0.354	3.059
	50	0.0005	0.047
	100	0.00019	0.008
	400	−0.00019	0.0007
	1,000	−0.00021	0.0003
Atallah algorithm	4	−0.19	2.322
	50	−0.007	1.566
	100	−0.450	9.433
	400	−0.460	9.503
	1,000	−0.463	9.573

The conic duality method is the fastest method overall. However, when only one starting point is considered, the SD of the relative error is high in comparison with that of other methods, indicating a low accuracy. On the other hand, if four or more starting points are considered, the method is able to consistently find the global optimal answer. In particular, four starting points would seem to be enough to find the optimal solution because no variations in the solutions are noted when more starting points are considered. For these reasons the c-d method with four starting points is considered to be the best optimization method implemented.

Discrete Geometric Methods

The same experiments were also conducted using the discrete geometric methods. Computational times for these are shown in Table 3, and the relative errors between the methods and the base method are displayed in Table 4.

Here some contradictory results were obtained. The Atallah algorithm requires more execution time than the exhaustive search method, even though the opposite could be expected, given the complexities of both methods. The reason is that to implement the former it is necessary to make various comparisons between points to check whether the geometrical conditions that make a point optimal are satisfied, whereas the latter has a straightforward implementation that does not require these types of comparisons. These geometrical conditions also affect the accuracy of the method. The way these conditions were implemented requires comparisons between float-point numbers, which carry precision issues yielding a higher error than they should. When the relative error of the Atallah algorithm for different values of N is compared with that for the c-d method, it is clear that this effect appears when 100 or more discretization points are used. For these, the distances obtained are higher than the ones obtained with the c-d method, yielding a negative

Table 5. Differences between rankings using the c-d method and exhaustive search with $N = 4$.

Seed	1st	2nd	3rd	4th–10th	≥11th
	. . . .(%).				
19	84.62	10.50	2.92	1.96	0.00
29	83.06	11.06	3.56	2.30	0.02
39	83.54	10.64	3.50	2.24	0.08
49	83.10	10.82	3.62	2.44	0.02
59	82.58	11.56	3.18	2.68	0.00
Mean	83.38	10.916	3.356	2.324	0.024
SD	0.772	0.417	0.297	0.265	0.033

mean value for the error. In addition, the SD increases considerably when going from 50 to 100 points, which implies a reduction in the reliability of the method.

On the other hand, the *exhaustive search* method approximates very well to the c-d method. For both types of ellipses, the relative error with use of 400 points or more is very similar to that obtained with the optimization method, taking half of the computational time. It is also important to note that use of more than 400 points does not improve the accuracy of the method by much, and it only implies an increase in the computational time.

For these reasons, the exhaustive search method with 400 points is considered to be the best proxy to the optimal Pompeiu-Hausdorff distance because it maintains accuracy while reducing the computational time considerably.

Quality of the Approximation for Ranking

Bear in mind that our ultimate goal is not to compute distances for their own sake but rather to compare distances between pairs of ellipses. As described in the Introduction, for the purpose of probability calculation, we want to assign each simulated fire to its nearest fire scar in the set of representative fire scars (ε). In this section, we analyze how the different approximation methods behave when used to rank a set of ellipses according to the Pompeiu-Hausdorff distance to another ellipse, with particular attention to how the presumptive closest ellipse is ranked.

We have seen that of the several methods implemented the one that could approximate the Pompeiu-Hausdorff distance the best was the c-d method with four starting points; however, the method takes an average of 0.991 second, which is a considerable amount of time if we want to compute several thousand calculations. It was also noted that the exhaustive search method was more efficient at the cost of accuracy, depending on the number of discrete points used (N).

To assess the differences in how these methods rank the representative fires, an experiment concerning 300 representative and 5,000 sampled fires for five different seeds was conducted. The c-d method and the exhaustive search approximation with different numbers of points were used to rank the representative fires for every set. Because the c-d method is the best bound on the Pompeiu-Hausdorff distance we have obtained, the ranking generated using this method is taken as the reference. Taking this into account, we computed the position that the ellipse ranked as the closest representative ellipse with the c-d method appeared on the ranking generated by the exhaustive search method. Results for the percentage of times the representative ellipse ranked as closest by the c-d method appears on each position of the exhaustive search ranking for different amount of discrete points (N) are shown in Tables 5–8. It is important to note that because the c-d method is computationally costly, only the first 20 ranked ellipses with the exhaustive search

Table 6. Differences between rankings using the c-d method and exhaustive search with $N = 50$.

Seed	1st	2nd	3rd	4th–10th	≥11th
			.......(%).......		
19	99.28	0.44	0.16	0.12	0.00
29	99.18	0.54	0.08	0.20	0.00
39	99.08	0.70	0.10	0.12	0.00
49	99.14	0.62	0.10	0.14	0.00
59	99.04	0.58	0.30	0.08	0.00
Mean	99.144	0.576	0.148	0.132	0.00
SD	0.093	0.096	0.090	0.044	0.00

Table 7. Differences between rankings using the c-d method and exhaustive search with $N = 100$.

Seed	1st	2nd	3rd	4th–10th	≥11th
			.......(%).......		
19	99.40	0.32	0.16	0.12	0.00
29	99.26	0.44	0.10	0.20	0.00
39	99.30	0.48	0.10	0.12	0.00
49	99.42	0.34	0.10	0.14	0.00
59	99.26	0.36	0.30	0.08	0.00
Mean	99.328	0.388	0.152	0.132	0.00
SD	0.077	0.069	0.087	0.044	0.00

Table 8. Differences between rankings using the c-d method and exhaustive search with $N = 400$.

Seed	1st	2nd	3rd	4th–10th	≥11th
			.......(%).......		
19	99.44	0.28	0.16	0.12	0.00
29	99.32	0.38	0.10	0.20	0.00
39	99.40	0.38	0.10	0.12	0.00
49	99.46	0.30	0.10	0.14	0.00
59	99.26	0.36	0.30	0.08	0.00
Mean	99.376	0.34	0.152	0.132	0.00
SD	0.084	0.047	0.087	0.044	0.00

method with $N = 400$ were taken into consideration to generate the c-d rankings.

Naturally, increasing the number of discrete points used for the exhaustive search method increases the accuracy of the ranking generated. This is shown by the percentage of times both methods yield the same representative fire as the closest one. Note that with use of more than 50 points for the exhaustive search method, <1% of the rankings yielded a closest ellipse different from the one returned by the c-d method.

Taking into consideration these results and the need for a fast and accurate methodology, we propose a multistage approach. In the first stage, representative fires are ranked using the exhaustive search method with $N = 4$ after which only a percentage of them are kept for further analysis. In this way, representative ellipses that are far away from the sampled fire are discarded using few computational resources. The remaining representative ellipses are ranked using the results obtained with the exhaustive search method with $N = 50$ as input for the c-d method starting points. Using the more accurate method for final ranking assures ~~a high level of approximation to the exact distance~~. Even though we cannot assure an exact fit, the level of approximation is high, while it reduces the time of computation considerably, making the multistage method viable, as opposed to the exact methods that need computational resources that might be excessive in some situations.

Table 9. Forest coverage for different number of representative ellipses, S_1.

No. of representative patterns	Mean	Minimum	Maximum	SD
		......(cells)......		
10	10.439	9.695	11.943	896.35
50	27.979	26.225	30.357	1,635.07
100	35.431	34.439	36.773	1,086.87
200	39.410	38.885	39.765	324.45
300	39.965	39.948	39.995	19.18
400	39.993	39.975	40.000	10.44
450	39.998	39.990	40.000	4.47
500	40.000	40.000	40.000	0

Representative Patterns

The set of representative patterns, ε, is the discretization of the space of possible fires that will constitute the scenarios. Hence, we want to work with a number of patterns that cover the forest fairly well (i.e., all fire cells covered by at least one representative pattern). To do this, we use a simple procedure for generating the representative patterns: we repeat the simulation of the S_1 representative patterns 10 times to generate candidate ε sets; the algorithm then selects the set that covers the most fire cells.

To assess its efficacy, an experiment in which this procedure is replicated 5 times for each value of S_1 is done. Statistics for this experiment are shown in Table 9.

Considering the coverage and the trade-off between accuracy and computational time, we conduct the remaining experiments using 200 representative patterns because this number provides reasonable coverage and is computationally expedient.

Probability Convergence

There are usually limited computational resources available, which necessitate a trade-off between speed and the number of simulations we can do. In turn, being able to generate only a limited number of simulations generates a statistical error, originating a second trade-off between number of runs and accuracy of the results.

We have shown that the execution time for one distance calculation between ellipses is sensitive to the method being used. Even though the multistage approach proposed for ranking calculation for each simulated fire is relatively fast, taking only 0.546 (seconds) for $|\varepsilon| = 200$, it would still add up to a considerable amount of time to simulate several thousand ellipses. This limits our ability to generate as many simulations as we desire in a reasonable time, especially if we wanted to consider a larger set of representative patterns.

Statistics of Estimands

The ultimate goal of our procedure is to estimate $|\varepsilon|$ probabilities, one for every representative pattern (200 in this particular analysis). Hence, there is a need to analyze each of these estimates separately, because they most likely will have different statistical characteristics and convergence behaviors. We will analyze these characteristics by looking at the conditional probability of burning, given that there is a fire

$$\mathbb{P}(\text{Burn}_i \mid \text{Fire}) = \frac{\text{Hits}_i}{N} \quad \forall_i \in \{1 \ldots |\varepsilon|\} \quad (11)$$

where N is the number of simulated fires in the sequence and Hits_i is the number of times a simulated (sampled) fire is assigned to the
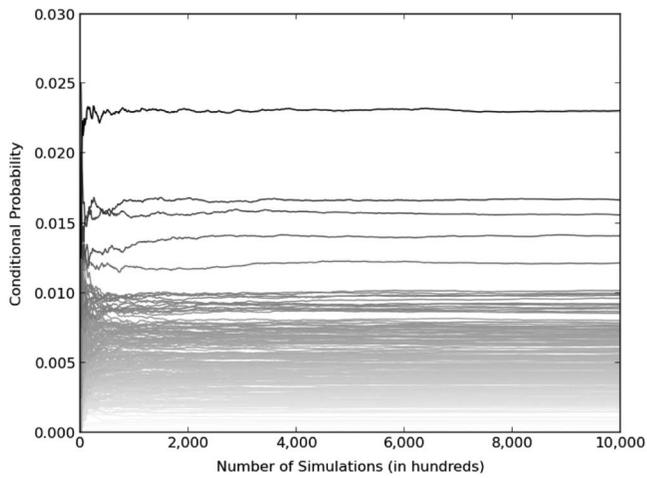
Figure 4. Probability of burning given that there is a fire for each representative pattern.

F4 ith pattern. Figure 4 shows the probabilities averaged over five independent fire sequences as a function of the number of simulated (sampled) fires; each sequence has 1,000,000 simulated fires. Naturally, the values are irregular at first but tend to stabilize as more and more simulations are considered. From simple inspection of the values given per iteration, one can say that not much is gained by simulating more than 200,000 fires. It is also seen that values for every probability stabilize around a single value, simplifying the analysis because statistics estimated at early stages can be used as *realistic* estimators, as opposed to a situation where divergence or temporary convergence was observed.

To support this result, we analyzed the SD of the probabilities based on elements introduced in Gelman and Rubin (1992) for the variance, studying the following statistics

A = average of m within-sequence standard deviations, each based on $n - 1$ *df*.

S = SD between the m sequence means, each based on n values of x.

where m is the number of sequences of fires simulated, n is the number of simulated fires in each sequence considered, and x is the value being estimated. To compute the statistics the first part of each sequence was dropped (ironically referred to as "the burn-in period"), leaving only the last 10,000 simulations. Of these, values given between intervals of 100 samples were used (i.e., 100 data points were considered for every calculation). By considering a constant number of samples, we are able to isolate the statistical characteristics of the last simulations from early variations in the sequences that do not contribute to having accurate information. It also allows us to calculate the statistical error based solely on the information retrieved by the SD of the 100 data points. Values for **F5** these statistics are presented in Figure 5, where they are displayed separately per quantile. The first quantile is the 40 least probable (of the 200) representative fire patterns. In this figure it is possible to detect a clear drop in all quantile in the average SD within sequences (A) between 20,000 and 200,000 simulations, after which SD reduction drops at a slower rate. This behavior is also seen for S, suggesting that probabilities tend to be independent of the random seed as more simulations are used, which is natural. Interestingly, higher probabilities require more samples to be as stable between sequences as low probabilities are, which could be troublesome if S

was high. However, S is on the order of 10^{-5} for all probabilities after only 20,000 simulations.

Based on this information, we can build a precision policy based only on the SD of the estimands. For example, let $\hat{\mu}_i$ be the estimated mean based on 100 data points (10,000 simulations) for the ith representative pattern and $\hat{\sigma}_i$ its estimated SD. Based on the central limit theorem, a $(1 - \alpha)\%$ confidence interval for the true mean is defined as follows

$$\left[\hat{\mu}_i - z_{1-\frac{\alpha}{2}} \cdot \frac{\hat{\sigma}_i}{\sqrt{n}}, \quad \hat{\mu}_i + z_{1-\frac{\alpha}{2}} \cdot \frac{\hat{\sigma}_i}{\sqrt{n}} \right] \tag{12}$$

where n is the number of simulations and $z_1 - (\alpha/2)$ is derived from the cumulative normal distribution function. We can impose the precision level we want (ω) with certain confidence by imposing conditions on the width of the confidence interval

$$2 \cdot z_{1-\frac{\alpha}{2}} \cdot \frac{\hat{\sigma}_i}{\sqrt{n}} \leq \omega \tag{13}$$

With this we can determine the statistical error that complies with the precision policy defined. In turn, this will allow us to determine the total number of replications needed to reach that error for all estimands. For instance, defining $\omega = 10^{-4}$ and $\alpha = 0,05$ implies that

$$\hat{\sigma}_i \leq \frac{10^{-4} \cdot \sqrt{100}}{2 \cdot 1,96} \approx 2,55 \cdot 10^{-4}$$

Which holds true for all $i \in \{1 \ldots |\varepsilon|\}$ after 50,000 simulations.

It would be desirable to impose even lower precision levels; however, this is not always possible due to computational concerns. Figure 6 **F6** shows the number of simulations needed to obtain different precision levels using $\alpha = 0.05$ and the computational time needed to compute them running 20 parallel processes. Computational times increase linearly among the number of simulations; however, the increase in precision does not follow the same trend. In fact, after 100,000 simulations we reached a precision of order 10^{-5}, which is reduced by 1 order of magnitude only after 800,000 simulations. Considering the computational resources needed to obtain this precision level, it seems unreasonable to simulate more than 100,000 ellipses.

Probability Vector Convergence

We can arrive at similar conclusions by analyzing the $|\varepsilon|$ probabilities as a whole, thinking of the vector of conditional probabilities retrieved after the jth iteration (simulated fire) as a probability distribution $P_j = (p_{j1}, p_{j2}, \ldots, p_{j|\varepsilon|})$, where p_{ji} is the conditional probability that the ith representative pattern burns, given that there is a fire for the ith representative pattern after j fires have been simulated. Clearly

$$\sum_{i=1}^{|\varepsilon|} p_{ji} = 1 \quad \forall j \tag{14}$$

allowing the use of measures of distance to assess similarity or divergence between probability distributions. A comprehensive review of metrics used to study probability convergence is presented in Gibbs and Su (2002). We use the *Hellinger distance*, given its properties and simplicity.

The Hellinger distance (Le Cam and Yang 2000) between two discrete probability distributions P and Q over a space Ω is defined as

(A) 1st Quantile

(B) 2nd Quantile

(C) 3rd Quantile

(D) 4th Quantile

(E) 5th Quantile
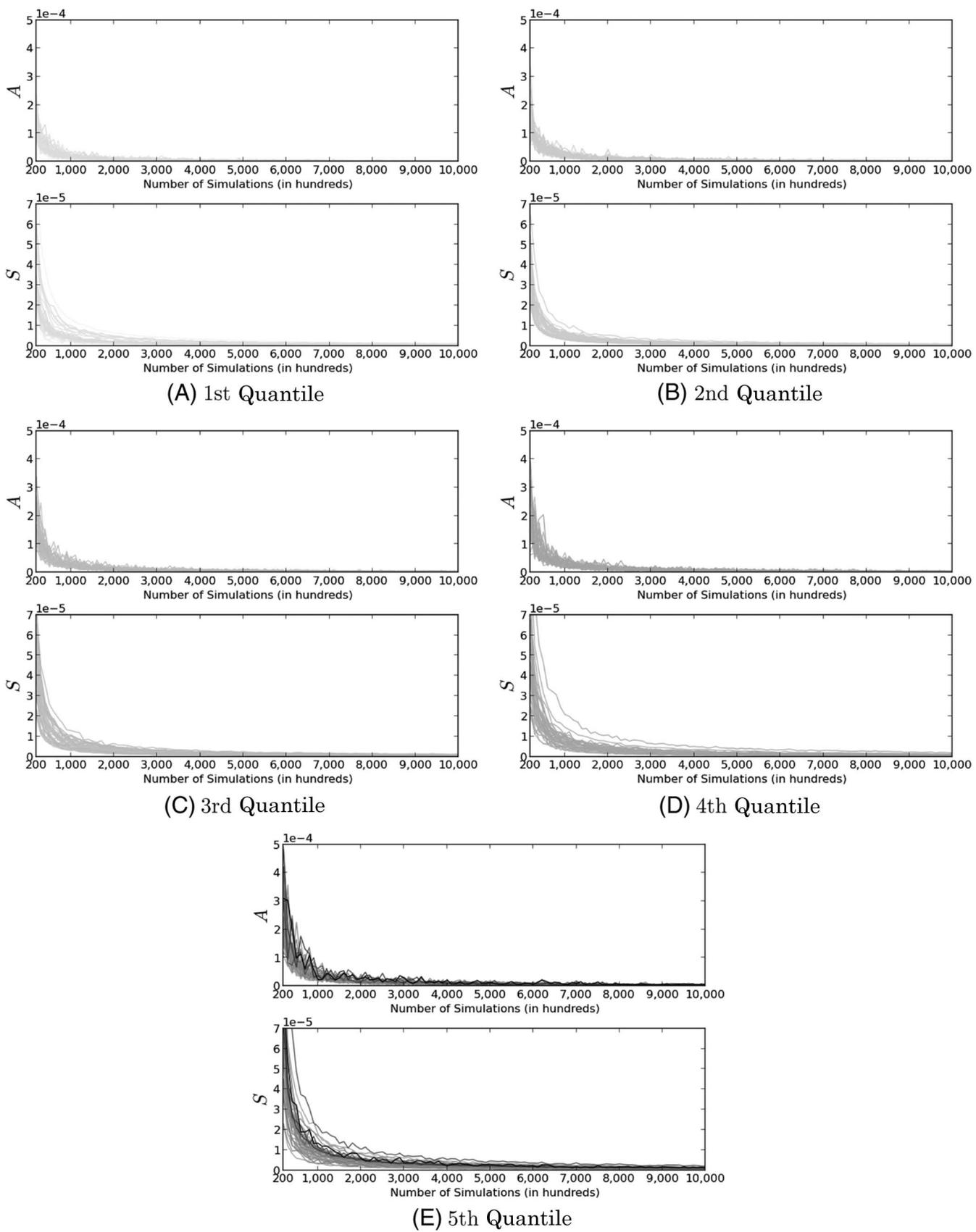
Figure 5. A and S for each representative pattern per quantile.
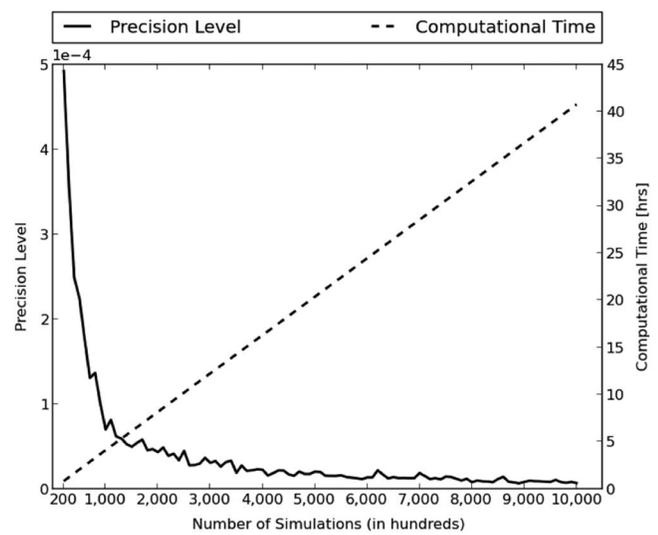
Figure 6. Precision level obtained and computational time per number of simulations.
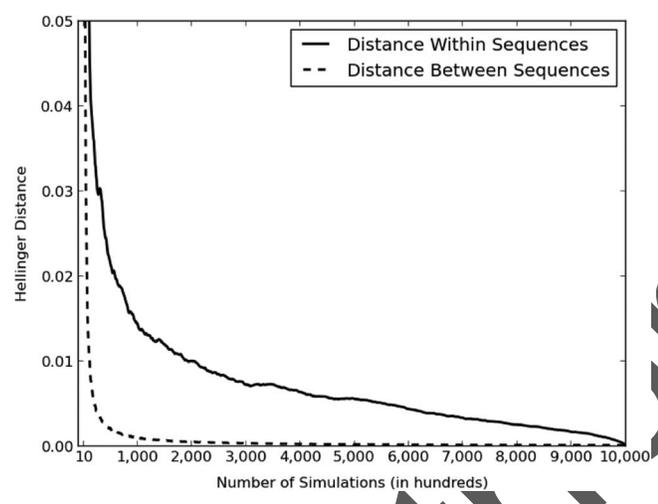
Figure 7. Hellinger distance between probability vectors for representative patterns.

$$h(P, Q) = \frac{1}{\sqrt{2}} \sqrt{\sum_{\omega \in \Omega} \left(\sqrt{P(\omega)} - \sqrt{Q(\omega)} \right)^2} \qquad (15)$$

It follows that (Gibbs and Su 2002)

$$0 \le h(P, Q) \le 1 \qquad (16)$$

We computed the distance every 1,000 simulations both between sequences and within sequences, taking the vector obtained using 1,000,000 simulations as a reference for the latter. Figure 7 displays the mean distance between all pairs of sequences and the distance within sequence as a function of the number of simulations. Similar behavior can be seen for both statistics. A clear drop in distance is observed between 1,000 and 100,000 simulations, after which the rate of reduction decreases.

After analyzing the estimated probabilities as a function of the number of iterations, both separately and as a vector, we can conclude that not much is gained by simulating more than 100,000 fires. However, even this many iterations may seem excessive, given that confidence intervals are relatively tight after only 20,000. In this context, it is not how many iterations are enough that is important, but how many iterations one can afford to compute, given the fire simulator in use and the available computational resources. Experiments with $|\varepsilon| = 300$ produced similar results.

Discussion and Conclusions

We have described a method of creating realistic fire scar scenarios for strategic planning as well as extensive computational experiments that describe its performance. Our algorithm for generating the fire-cell resolution fire scar scenarios is as follows

1. Find S_1 fire scars by doing S_1 simulations n times to form the set of fire cell sets (i.e., fire scars) ε that provides the best coverage as described

2. Run S_2 simulations and assign each fire to its nearest fire scar in ε.

3. The probability associated with each member of ε is the number of assignments divided by S_2.

When the algorithm is used in practical settings, experiments may be needed to set the parameters for any particular forest; however, we have shown that for a hypothetical forest that is representative of the boreal forest region of Canada, $n = 10$, $S_1 = 200$, and $S_2 = 5,000$ works well and $S_2 = 20,000$ works very well. Generation of 200 scenarios requires a few minutes on a laptop computer when the distances needed in step 3 are computed using a two-stage heuristic. Consequently, larger values of S_1 and S_2 are tractable in some applications. We have developed, tested, and illustrated the use of a methodology for generating discrete sets of stochastic forest and wildland fire scar scenarios by using a very simple fire ignition and growth model and applying it to a simple hypothetical forest, but our methodology can readily be enhanced and applied to evaluate strategies for managing risk on real flammable landscapes.

We assumed that the locations at which fires might ignite are uniformly distributed across our forest, but fire ignition processes are influenced by fuel, weather demographics, and land-use patterns. Brillinger et al. (2003, 2006) used generalized mixed-effects methods to develop spatially explicit fire ignition models for portions of the states of Oregon and California, respectively, and their approach could be incorporated in our methodology. We used a very simple elliptical model, but it could readily be replaced by more complex models that account for variations in fire growth across the landscape. Preisler et al. (2004) built on the methods developed in Brillinger et al. (2003) to develop a spatially and temporally explicit nonparametric regression model that can be used to predict the probability that a fire which occurs in a specific 1×1 km cell in their study area in the state of Oregon would become large (40.5 ha or more) or the conditional probability that a large fire will occur in a cell. Ziesler et al. (2013) used fire scar data from Yellowstone National Park to develop a conditional burn probability methodology that one could use to predict the conditional probability that a fire will grow to fill a specified set of cells on a landscape that might be used to model the generation of fire scars in a BP model.

One could, of course, use complex fire spread models (e.g., variants of the Huygens type model developed by Richards [1995]) to model fire growth. The probability that a fire will escape initial attack and its final size depend, of course, on both the productivity of the initial and extended attack systems and the suppression models (e.g., Fried and Fried 1996). The large-fire simulation system

AQ: F

AQ: G

AQ: H

F7

(FSim) described by Finney et al. (2011) includes ignition, suppression, and fire growth components and could be used to generate the type of fire scar patterns on which our methodology is based.

We have focused on the generation of single-fire scenarios to simplify our analysis, and because most forest management units in the boreal forest region of the province of Ontario seldom experience more than one large fire each year, a revised version of our methodology could be expanded to deal with the possibility that more than one fire might occur during each period. The challenge is to develop a suitable method of measuring how similar a specific scenario is to the initial set of baseline scenarios to which it must be compared. We are working in continuous space, and it would be challenging to extend our current methods to develop a continuous space methodology that deals with multiple fire scenarios. One could, however, discretize the landscape into small cells as others do and compare patterns in terms of the number of cells in common.

We note that the use of the stochastic fire scar patterns that can be generated using our methodology need not be restricted to the development of strategic forest management planning models. Once the patterns are developed they could be used to easily answer some of the questions that traditional burn probability models are used to answer. Suppose, for example, the forest management unit contained a home and one wanted to estimate the probability that home will burn. One could identify all of the S_1 fire scar scenarios that intersect that home and the probability that the home will be destroyed could quickly be determined by summing the probabilities of the scenarios that intersect the home.

But our fire scar scenarios can also be used to answer questions that are not so readily answered using traditional BP models. Suppose one wanted to locate two conservation areas on a flammable forest landscape. One could use a modified version of the traditional BP model inasmuch as one could tally not the probability that each pixel or cell on the landscape would burn but rather the number of times both of those conservation areas burned. If one wanted to evaluate alternative locations, one would have to rerun the burn probability model for each of the conservation area configurations one would like evaluated.

Our burn scar patterns could, however, be much more easily used to evaluate alternative conservation area configurations. One could propose a configuration, identify all of the single fire burn scar scenarios that burned both of the conservation areas, and sum their probabilities to estimate the probability that both would burn. One could identify other configurations and just as quickly also evaluate them. One could also easily answer more complex questions such as, the probability that neither would burn, that area 1 would burn and area 2 would not, or that area 2 would burn and area 1 would not burn, all using the same set of fire scar scenarios.

In closing, we note that the need to generate stochastic fire scars and incorporate them in stochastic programming models extends far beyond the strategic forest management planning under uncertainty due to fire that motivated the development of our methodology. Forest and wildland fire managers have to deal with considerable uncertainty when they attempt to resolve their strategic, tactical, and operational decisionmaking problems, and we recognize that they have yet to exploit important advances in stochastic programming. Initial efforts to apply stochastic programming methods to fire and forest management focused on strategic planning problems (e.g., Gassmann 1989, Boychuk and Martell 1996), but there have been several recent attempts to apply stochastic programming methods to operational fire problems (e.g., Ntaimo et al. 2012, 2013, Lee et al.

2013). Given the very broad and diverse decisionmaking problems that confront forest and wildland fire managers, there is a need to further explore what stochastic programming has to offer and to evaluate how well specific approaches perform when they are applied to specific classes of forest and wildland fire management decisionmaking problems.

Endnote

1. White and Pickett (1985) defined a disturbance as "any relatively discrete event in time that disrupts ecosystem, community, or population structure and changes resources, substrate availability, or the physical environment." It is important to note that despite the fact that the term "disturbance" has negative connotations and natural disturbance processes can and often do have destructive impacts on public safety, property and forest resources, natural disturbance processes are natural forest ecosystem processes and essential to ecosystem health.

Literature Cited

AGER, A.A., N.M. VALLIANT, M.A. FINNEY, AND H.K. PREISLER. 2012. Analyzing wildfire exposure and source-sink relationships on a fire prone forest landscape. *For. Ecol. Manage.* 267:271–283.

ATALLAH, M.J. 1983. A linear time algorithm for the Hausdorff distance between convex polygons. *Inform. Process. Lett.* 17(4):207–209.

ATALLAH, M.J., C.C. RIBEIRO, AND S. LIFSCHITZ. 1991. A linear time algorithm for the computation of some distance functions between convex polygons. *RAIRO Rech. Oper.* 25(4):413–424.

BEN-TAL, A., L. EL GHAOUI, AND A. NEMIROVSKI. 2009. *Robust optimization.* Princeton Univ. Press, Princeton, NJ. 576 p.   AQ: I

BOYCHUK, D., AND D. MARTELL. 1996. A multistage stochastic programming model for sustainable forest-level timber supply under risk of fire. *For. Sci.* 42(1):10–26.

BRILLINGER, D., H. PREISLER, AND J. BENOIT. 2003. Risk assessment: A forest fire example. P. 177–196 in *Statistics and science: A festschrift for Terry Speed,* Vol. 40: Lecture Notes Monograph Series. Institute of Mathematical Statistics, Beachwood, OH.

BRILLINGER, D.R., H.K. PREISLER, AND J.W. BENOIT. 2006. Probabilistic risk assessment for wildfires. *Environmetrics* 17(6):623–633.

EBERLY, D. 2003. Information about ellipses. *Magic Software, Inc.* Available online at www.magicsoftware.com; last accessed Dec. 3, 2014.   AQ: J

EDELSBRUNNER, H. 1985. Computing the extreme distances between two convex polygons. *J. Algorithms* 6(2):213–224.

FINNEY, M.A., C.W. MCHUGH, I.C. GRENFELL, K.L. RILEY, AND K.C. SHORT. 2011. A simulation of probabilistic wildfire risk components for the continental United States. *Stochastic Environ. Res. Risk Assess.* 25(7):973–1000.

FRIED, J.S., AND B.D. FRIED. 1996. Simulating wildfire containment with realistic tactics. *For. Sci.* 42(3):267–281.

GASSMANN, H.I. 1989. Optimal harvest of a forest in the presence of uncertainty. *Can. J. For. Res.* 19(10):1267–1274.

GASSMANN, H., AND A. IRELAND. 1995. Scenario formulation in an algebraic modelling language. *Ann. Oper. Res.* 59(1):45–75.

GELMAN, A., AND D.B. RUBIN. 1992. Inference from iterative simulation using multiple sequences. *Stat. Sci.* p. 457–472.

GIBBS, A.L., AND F.E. SU. 2002. On choosing and bounding probability metrics. *Int. Stat. Rev.* 70(3):419–435.

HANEWINKEL, M., S. HUMMEL, AND A. ALBRECHT. 2011. Assessing natural hazards in forestry for risk management: A review. *Eur. J. For. Res.* 130(3):329–351.

KING, A., AND S. WALLACE. 2010. *Modelling with stochastic programming.* Springer, New York. 173 p.   AQ: K

KONOSHIMA, M., H.J. ALBERS, C.A. MONTGOMERY, AND J.L. ARTHUR. 2010. Optimal spatial patterns of fuel management and timber harvest with fire risk. *Can. J. For. Res.* 40(1):95–108.

LE CAM, L., AND G.L. YANG. 2000. *Asymptotics in statistics: Some basic concepts.* Springer, New York. 287 p.   AQ: L

LEE, Y., J. FRIED, H. ALBERS, AND R. HAIGHT. 2013. Deploying initial attack resources for wildfire suppression: Spatial coordination, budget constraints, and capacity constraints. *Can. J. For. Res.* 43:56–65.

MARTELL, D.L. 1980. The optimal rotation of a flammable forest stand. *Can. J. For. Res.* 10(1):30–34.

MILLER, C., M.A. PARISIEN, A.A. AGER, AND M.A. FINNEY. 2008. Evaluating spatially-explicit burn probabilities for strategic fire management planning. P. 245–252 in *Modelling, monitoring and management of forest fires, Vol. 119: Wit transactions on ecology and the environment,* DeLas-Heras, J., C.A. Brebbia, D. Viegas, and V. Leone (eds.). Wessex Institute of Technology, Wessex, UK.

NAVON, D.I. 1971. *Timber resources allocation method a long-range planning method for commercial timber lands under multiple use management.* USDA For. Serv., Res. Pap. PSW-70, Pacific Northwest Research Station, Berkeley, CA. 22 p.

NTAIMO, L., J.A. GALLEGO-ARRUBLA, J. GAN, C. STRIPLING, J. YOUNG, AND T. SPENCER. 2013. A simulation and stochastic integer programming approach to wildfire initial attack planning. *For. Sci.* 59:105–117.

NTAIMO, L., J. GALLEGO ARRUBLA, C. STRIPLING, J. YOUNG, AND T. SPENCER. 2012. A stochastic programming standard response model for wildfire initial attack planning. *Can. J. For. Res.* 42:987–1001.

O'ROURKE, J., C.-B. CHIEN, T. OLSON, AND D. NADDOR. 1982. A new linear algorithm for intersecting convex polygons. *Comput. Graphics Image Process* 19(4):384–391.

PREISLER, H. K., D.R. BRILLINGER, R.E. BURGAN, AND J.W. BENOIT. 2004. Probability based models for estimation of wildfire risk. *Int. J. Wild. Fire* 13(2):133–142.

REED, W.J. 1984. The effects of the risk of fire on the optimal rotation of a forest. *J. Environ. Econ. Manage.* 11(2):180–190.

REED, W.J., AND D. ERRICO. 1986. Optimal harvest scheduling at the forest level in the presence of the risk of fire. *Can. J. For. Res.* 16(2):266–278.

RICHARDS, G.D. 1995. A general mathematical framework for modeling 2-dimensional wildland fire spread. *Int. J. Wildl. Fire* 5(2):63–72.

ROCKAFELLAR, R., AND R. WETS. 1998. *Variational analysis, Vol. 317, Grundlehren der Mathematischen Wissenschaft*, 3rd printing, 2009 ed. Springer, New York.

ROUTLEDGE, R.D. 1980. The effect of potential catastrophic mortality and other unpredictable events on optimal forest rotation policy. *For. Sci.* 26(3):389–399.

SCOTT, J.H., D.J. HELMBRECHT, S.A. PARKS, AND C. MILLER. 2012. Quantifying the threat of unsuppressed wildfires reaching the adjacent wildland-urban interface on the Bridger-Teton National Forest, Wyoming, USA. *Fire Ecol.* 8(2):125–142.

THOMPSON, M.P., AND D.E. CALKIN. 2011. Uncertainty and risk in wildland fire management: A review. *J. Environ. Manage.* 92(8):1895–1909.

THOMPSON, M.P., J. SCOTT, J.D. HAIDEN, AND J.W. GILBERTSON-DAY. 2013. A polygon-based modeling approach to assess exposure of resources and assets to wildfire. *Nat. Hazards* 67(2):627–644.

VALSTA, L.T. 1992. A scenario approach to stochastic anticipatory optimization in stand management. *For. Sci.* 38(2):430–447.

VAN WAGNER, C. 1969. A simple fire growth model. *For. Chron.* 45(2):103–104.

WÄCHTER, A., AND L.T. BIEGLER. 2006. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.* 106(1):25–57.

WALTZ, R.A., AND J. NOCEDAL. 2003. *KNITRO user's manual.* Northwestern Univ., Tech. Rep. OTC-2003/5, Evanston, IL. 44 p.

WEI, Y., D. RIDEOUT, AND A. KIRSCH. 2008. An optimization model for locating fuel treatments across a landscape to reduce expected fire losses. *Can. J. For. Res.* 38(4):868–877.

WHITE, P.S., AND S. PICKETT. 1985. Natural disturbance and patch dynamics: An introduction. P. 3–15 in *The ecology of natural disturbance and patch dynamics,* Pickett, S.T.A., and P.S. White (eds). Academic Press, New York.

ZIESLER, P.S., D.B. RIDEOUT, AND R. REICH. 2013. Modelling conditional burn probability patterns for large wildland fires. *Int. J. Wild. Fire* 22(5):579–587.

Appendix: Details of Conic Duality

In this Appendix we provide details of the conic duality method. Introducing $z \in \Re$ such that

$$z \geq \left\| (\bar{x}_1, \bar{y}_1), (\bar{x}_2(t), \bar{y}_2(t)) \right\|_2 \tag{17}$$

$$\Rightarrow \left((\bar{x}_1 - \bar{x}_2(t)) \quad (\bar{y}_1 - \bar{y}_2(t)) \quad z \right) \in \mathscr{L}^3 \tag{18}$$

problem 8 can be expressed as a conic maximization-minimization problem

$$\max_{(\bar{x}_2(t), \bar{y}_2(t)) \in \bar{E}_2} \min_{(\bar{x}_1, \bar{y}_1, z)} z \tag{19}$$

s.t.

$$\left(\frac{\bar{x}_1}{a_1} \quad \frac{\bar{y}_1}{b_1} \quad 1 \right) \in \mathscr{L}^3$$

$$\left((\bar{x}_1 - \bar{x}_2(t)) \quad (\bar{y}_1 - \bar{y}_2(t)) \quad z \right) \in \mathscr{L}^3$$

The minimization problem (20) is the one of interest, because it is necessary to transform it to end up with a maximization problem, avoiding the calculation of the discrete benchmark method.

$$\min_{(\bar{x}_1, \bar{y}_1, z)} z \tag{20}$$

s.t.

$$\left(\frac{\bar{x}_1}{a_1} \quad \frac{\bar{y}_1}{b_1} \quad 1 \right) \in \mathscr{L}^3$$

$$\left((\bar{x}_1 - \bar{x}_2(t)) \quad (\bar{y}_1 - \bar{y}_2(t)) \quad z \right) \in \mathscr{L}^3$$

This is equivalent to

$$\min_{(\bar{x}_1, \bar{y}_1, z)} (0 \quad 0 \quad 1) \cdot \begin{pmatrix} \bar{x}_1 \\ \bar{y}_1 \\ z \end{pmatrix}$$

s.t.

$$\begin{pmatrix} \frac{1}{a_1} & 0 & 0 \\ 0 & \frac{1}{b_1} & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} \bar{x}_1 \\ \bar{y}_1 \\ z \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \in \mathscr{L}^3 \quad (\gamma 1, \gamma 2, \gamma 3) \tag{21}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \bar{x}_1 \\ \bar{y}_1 \\ z \end{pmatrix} - \begin{pmatrix} \bar{x}_2(t) \\ \bar{y}_2(t) \\ 0 \end{pmatrix} \in \mathscr{L}^3 \quad (\omega_1, \omega_2, \omega_3)$$

AQ: M

where γ_1, γ_2, γ_3, ω_1, ω_2, and ω_3 are the corresponding dual variables.

Applying conic duality to the previous problem, the following representation is obtained

$$\max_{\gamma_{1..3},\, \omega_{1..3}} \quad -\gamma_3 + \bar{x}_2(t) \cdot \omega_1 + \bar{y}_2(t) \cdot \omega_2$$

s.t.

$$(\gamma_1, \gamma_2, \gamma_3) \in \mathscr{L}^3 \tag{22}$$

$$(\omega_1, \omega_2, \omega_3) \in \mathscr{L}^3$$

$$\begin{pmatrix} \dfrac{1}{a_1} \cdot \gamma_1 \\[2mm] \dfrac{1}{b_1} \cdot \gamma_2 \\[2mm] 0 \end{pmatrix} + \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Using this last result, it is possible to transform 1 to the following maximization problem

$$\max_{t,\, \gamma_{1..3},\, \omega_{1..3}} \quad -\gamma_3 + \bar{x}_2(t) \cdot \omega_1 + \bar{y}_2(t) \cdot \omega_2$$

s.t.

$$(\gamma_1, \gamma_2, \gamma_3) \in \mathscr{L}^3$$

$$(\omega_1, \omega_2, \omega_3) \in \mathscr{L}^3$$

$$\begin{pmatrix} \dfrac{1}{a_1} \cdot \gamma_1 \\[2mm] \dfrac{1}{b_1} \cdot \gamma_2 \\[2mm] 0 \end{pmatrix} + \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{23}$$

$$\bar{x}_2(t) = e_x + f_x \cdot \cos t + g_x \cdot \sin t$$

$$\bar{y}_2(t) = e_y + f_y \cdot \cos t + g_y \cdot \sin t$$

where

$$e_x = (xc_2 - xc_1) \cdot \cos\phi_1 + (yc_2 - yc_1) \cdot \sin\phi_1 \tag{24}$$

$$f_x = a_2 \cdot (\cos\phi_1 \cdot \cos\phi_2 + \sin\phi_1 \cdot \sin\phi_2) \tag{25}$$

$$g_x = b_2 \cdot (\sin\phi_1 \cdot \cos\phi_2 - \cos\phi_1 \cdot \sin\phi_2) \tag{26}$$

$$e_y = (yc_2 - yc_1) \cdot \cos\phi_1 + (xc_1 - xc_2) \cdot \sin\phi_1 \tag{27}$$

$$f_y = a_2 \cdot (\cos\phi_1 \cdot \sin\phi_2 - \sin\phi_1 \cdot \cos\phi_2) \tag{28}$$

$$g_x = b_2 \cdot (\sin\phi_1 \cdot \sin\phi_2 + \cos\phi_1 \cdot \cos\phi_2) \tag{29}$$

Moreover, knowing that $\cos^2 t + \sin^2 t = 1$, it is possible to introduce $\alpha, \beta \in \mathfrak{R}$ such that $\alpha^2 + \beta^2 = 1$. With this, we obtain the final representation for the original problem

$$\max_{\alpha,\, \beta,\, \gamma_{1..3},\, \omega_{1..3}} \quad -\gamma_3 + (e_x + f_x \cdot \alpha + g_x \cdot \beta) \cdot \omega_1$$
$$+ (e_y + f_y \cdot \alpha + g_y \cdot \beta) \cdot \omega_2$$

s.t. $$\tag{30}$$

$$(\gamma_1, \gamma_2, \gamma_3) \in \mathscr{L}^3$$

$$(\omega_1, \omega_2, \omega_3) \in \mathscr{L}^3$$

$$\begin{pmatrix} \dfrac{1}{a_1} \cdot \gamma_1 \\[2mm] \dfrac{1}{b_1} \cdot \gamma_2 \\[2mm] 0 \end{pmatrix} + \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

$$\alpha^2 + \beta^2 = 1$$