



Estimation of wildfire wind conditions via perimeter and surface area optimization[☆]

Li Tan^{a,*}, Raymond A. de Callafon^a, Jessica Block^b, Daniel Crawl^b, Tolga Çağlar^b, İlkay Altıntaş^b

^a Department of Mechanical and Aerospace Engineering University of California San Diego, La Jolla, CA, USA

^b San Diego Supercomputer Center University of California San Diego, La Jolla, CA, USA

ARTICLE INFO

Keywords:

Wildfire
FARSITE
Uncertainty
Interpolation
Polygon
Surface area
Gradient-free optimization

ABSTRACT

This paper shows that the prediction capability of wildfire progression can be improved by estimation of a single prevailing wind vector parametrized by a wind speed and a wind direction to drive a wildfire simulation created by FARSITE. Estimations of these wind vectors are achieved in this work by a gradient-free optimization via a grid search that compares wildfire model simulations with measured wildfire perimeters, where noisy observations are modeled as uncertainties on the locations of the vertices of the measured wildfire perimeters. Two optimizations are established to acquire the optimal wind speed and wind direction. To formulate a perimeter optimization, an uncertainty-weighted least-squares error is computed between the vertices of the simulated and measured wildfire perimeter. The challenge in this approach is to match the number of vertices on the simulated and measured wildfire perimeter via interpolation of perimeter points and their uncertainties. For a surface area optimization, an uncertainty-weighted surface area error is introduced to capture the surface of the union minus the intersection of the simulated and measured wildfire perimeter. The challenge in this approach is to formulate a surface area error, weighted by the uncertainties on the locations of the vertices of the measured wildfire perimeter. The optimization in this paper is based on an iterative refinement of a grid of the wind vector and provides robustness to intermittent erroneous results produced by FARSITE, while allowing parallel execution of wildfire model calculations. This paper is an extension of the work in Tan et al., (2021). Results on wind vector estimation are illustrated on two historical wildfire events: the 2019 Maria Fire that burned south of the community of Santa Paula in the area of Somis, CA, and the 2019 Cave Fire that started in the Santa Ynez Mountains of Santa Barbara County.

1. Introduction

With the increased and inevitable occurrence of wildfires, more accurate and responsive prediction of the wildfire propagation is important for resource allocation in fire fighting efforts. The wildfire growth modeling software FARSITE is widely used by the U.S Forest Service to simulate the propagation of wildfires [1], and is characterized by the ability to estimate the wildfire propagation under heterogeneous conditions of terrain, fuels and weather. Crucial source of information in the modeling of fire progression is a single wind vector characterized by average wind speed and wind direction that determine the overall direction and rate of spread of the wildfire. This paper is an extension of the work in [2] by adding the ability to formulate an uncertainty-weighted wildfire surface coverage error to estimate the wind vector of

wind speed and wind direction, which is explained in Section 3.2. The numerical results are summarized in Section 5.

The prediction of the growth of wildfires has received a considerable amount of attention in the literature. Rothermel introduced the mathematical model for predicting fire spread [3], and experiments have been conducted to analyze the influence of fuel and weather on the spread of wildfires [4]. Further steps in the study of the wildfire behavior were achieved by adjusting model prediction using real-time data via data assimilation techniques [5–7]. Data assimilation by combining FARSITE and an ensemble Kalman filter has been done in earlier work [8–11] demonstrating an improvement in accuracy of wildfire prediction. The availability of unmanned aerial vehicles to better monitor large-scale

[☆] work is supported by WIFIRE commons and funded by National Science Foundation 2040676 under the Convergence Accelerator Program.

* Corresponding author.

E-mail addresses: ltan@eng.ucsd.edu (L. Tan), callafon@eng.ucsd.edu (R.A. de Callafon), jblock@ucsd.edu (J. Block), crawl@ucsd.edu (D. Crawl), tcaglar@ucsd.edu (T. Çağlar), altintas@ucsd.edu (I. Altıntaş).

<https://doi.org/10.1016/j.jocs.2022.101633>

Received 15 October 2021; Received in revised form 15 February 2022; Accepted 19 February 2022

Available online 26 February 2022

1877-7503/© 2022 Elsevier B.V. All rights reserved.

wildfire [12,13] has further enhanced the capabilities of data-driven wildfire modeling.

As mentioned in [4], among the numerous factors that can affect the spread of the wildfire, wind speed and wind direction play the critical roles. Unfortunately, wind conditions are available only from sparsely placed weather stations. Detailed studies are available on learning the (non-linear) relationship between the properties of the fuel and the wildfire progression [14–16], but often only limited information on wind speed and wind direction can be used. This means that the quality of the prediction is extremely dependent on the quality of an empirical estimate of the wind conditions obtained from geometrically spaced weather station. In reality, information of the actual wind conditions at the boundary of the wildfire is unavailable due to limited number of weather stations and the turbulent atmosphere caused by wildfire. As a result, significant and compounding errors can occur in the prediction of the wildfire propagation. A first step is to estimate the best initial wind conditions before any data assimilation procedure. In this situation, the error caused by an erroneous measurement of the wind conditions can be reduced, and the accuracy of the prediction by data assimilation techniques can be greatly improved.

A gradient-free optimization via a grid search is used in this work to provide an estimate of the single wind vector of wind speed and wind direction fed to FARSITE with the objective to improve the prediction of wildfire progression. The gradient-free optimization via an iterative grid search refines a grid of wind speed and wind direction to select the best wind vector based on a loss function that compares wildfire model simulations with noisy observations of the wildfire perimeters. Since each grid point provides an independent wildfire simulation, the computations can be executed in parallel and also provide robustness to possible erroneous perimeter produced by FARSITE under certain wind vector. To formulate the loss function, it is first shown that noisy observations can be modeled as uncertainties on the locations of the vertices of a measured wildfire perimeter. Secondly, it is shown that an uncertainty-weighted error can be computed between the vertices of a simulated and a measured wildfire perimeter.

In this paper two different uncertainty-weighted errors are formulated for the estimation of the single wind vector: a perimeter and surface area based. For the perimeter optimization, a skew compensated and uncertainty weighted least-squares error is computed between the vertices of a simulated and a measured wildfire perimeter. To be able to compute this perimeter error, it is shown that a linear interpolation of the perimeter is used to guarantee that the skew compensated weighted least-squares error can always be computed. Furthermore, compared to an ordinary weighted least-squares error, the weighting in the skew compensated weighted least-squares computation is adjusted to account for unevenly distributed polygons to allow an evenly distributed weighting of the complete wildfire perimeter. The surface area optimization captures the wildfire surface area error defined by the union minus the intersection of a simulated and a measured wildfire perimeter. By using the fact that the surface area of a closed polygon can be calculated as the signed sum of triangular sub-polygons [17,18], it is shown how to compute an uncertainty-weighted surface area error. The weighting is again determined by the uncertainties on the vertices of the measured wildfire perimeter and the computational process of the weighted surface area error is simple and fast.

The paper is organized as follows. Section 2 presents the model of the polygon data along with the uncertainties on the vertices of a wildfire perimeter. Following the uncertainty characterization, Section 3 presents the computations of the skew compensated uncertainty-weighted least-squares error and the uncertainty-weighted surface area error. Section 4 outlines the parallel gradient-free optimization via a grid search based on refining a grid of wind speed and wind direction to estimate the optimal wind vector. Section 5 shows the numerical results for the estimation of the wind vector for two use cases of wildfires in California: the 2019 Maria Fire that burned south of the community of Santa Paula and the 2019 Cave Fire that started in the Santa Ynez Mountains of Santa Barbara County. Conclusions are summarized in Section 6.

2. Wildfire perimeter and uncertainty

A wildfire may cover multiple disjoint burned areas. For simplicity of the analysis presented in this paper, the notion of wildfire progression is characterized by a wildfire perimeter that is considered to be a single closed polygon. The analysis presented here can be applied to each of the closed-polygons in case a wildfire does cover multiple disjoint burned areas. The single closed polygon describing the wildfire perimeter is an ordered sequence of N vertices and N piece-wise linear line segments. The vertices of the approximated polygon are located by the Eastern and Northern coordinate pairs $(e(k), n(k))$, $k = 1, 2, \dots, N$.

2.1. Uncertainty characterization

Measurements of the wildfire perimeters can be a combined data collection effort. The resolution and spacing of the measured vertices are determined by data from satellite imagery, aerial surveillance or manually mapped observations [19]. Therefore, it is important to consider a two-dimensional (2D) uncertainty for each vertex of the closed polygon that describes the measured wildfire perimeter. The general description of the 2D uncertainty on a vertex $(e(k), n(k))$ is a rotated ellipse, where the semi-major axis $a(k)$, semi-minor axis $b(k)$, and the rotation angle $\alpha(k)$ collectively reflect the variance in the horizontal direction and vertical direction. Such detailed information may not be available and therefore the uncertainty on a vertex $(e(k), n(k))$ is expressed by a circle centered at this vertex with a radius $r(k)$, where the value of $r(k)$ is proportional to the uncertainty of the vertex on the polygon.

However, it is very likely that a measured perimeter comes with no additional uncertainty characterization. In that case, the assumption is made that the uncertainty on each vertex is proportional to the (smallest) distance to the neighboring vertex on the polygon. The reason for that is the vertices are more likely to have large uncertainties for sporadic measurements with a large distance between the vertices. Formally this uncertainty is described by

$$\begin{aligned} r(k) &= \max(\min(l(k), l(k-1)), r_{\min}) \\ l(k) &= \sqrt{(e(k+1) - e(k))^2 + (n(k+1) - n(k))^2} \end{aligned} \quad (1)$$

for $k = 1, 2, \dots, N$, where $r(k)$ is the assumed uncertainty, $l(k)$ is the distance between neighboring vertices $(e(k+1), n(k+1))$ and $(e(k), n(k))$, and $r_{\min}$ is a user-defined minimum value of uncertainty radius. The value of $r_{\min}$ is used to avoid the condition in which two adjacent vertices are extremely close to each other, and can be determined by the accuracy of measuring method used to acquire the polygon of the wildfire perimeter. An illustration of the uncertainty assignment for a measured wildfire perimeter is given in Fig. 1.

2.2. Perimeter interpolation

With the spread of a wildfire, the corresponding closed polygon describing the measured wildfire perimeter commonly becomes larger and the number N_m of vertices of the measured closed polygon $(e_m(k_m), n_m(k_m))$, $k_m = 1, 2, \dots, N_m$ increases accordingly. Similarly, the number of vertices N_s on a simulated wildfire perimeter $(e_s(k_s), n_s(k_s))$, $k_s = 1, 2, \dots, N_s$ obtained with fire modeling software such as FARSITE will also increase, but in general $N_m \neq N_s$. The resolution of the simulated vertices is determined by the fire modeling software FARSITE and typically in the order of 30 m. Next to difference in number of vertices, the ordering of the vertices $(e_m(k_m), n_m(k_m))$, $k_m = 1, 2, \dots, N_m$ of the measured fire perimeter and $(e_s(k_s), n_s(k_s))$, $k_s = 1, 2, \dots, N_s$ are not the same and a direct comparison between a pair of vertices $(e_m(k_m), n_m(k_m))$ and $(e_s(k_s), n_s(k_s))$ would lead to erroneous results.

A direct comparison of a measured vertex $(e_m(k_m), n_m(k_m))$ and a simulated vertex $(e_s(k_s), n_s(k_s))$ is especially important if a (weighted) least-squares error based on vertices needs to be formulated. To anticipate the notion of an uncertainty weighted least-squares error, it

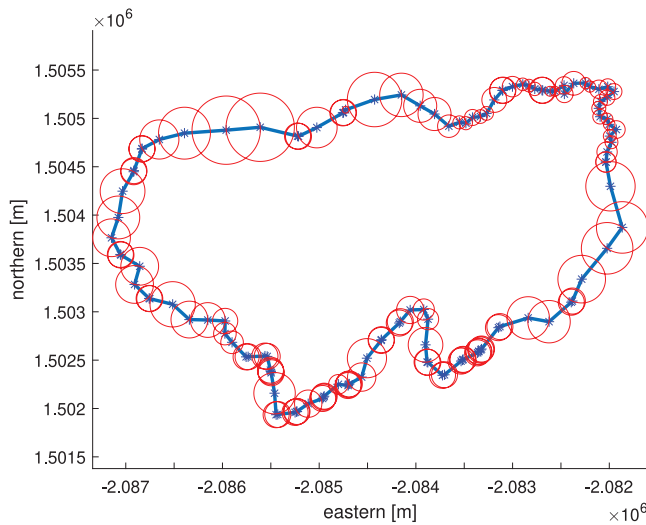


Fig. 1. Assignment of uncertainty radii $r(k)$ (red circles) on a measured wildfire perimeter with vertices $(e(k), n(k))$ (blue stars) and the resulting closed polygon (blue lines).

is shown how to perform an interpolation on the wildfire perimeter to create $N_m = N_s$ and therefore an equal number of N_m and N_s of vertices on the measured and the simulated closed polygon. The solution to this problem is to first interpolate one of the fire perimeters to the same or higher number $N = \max(N_m, N_s)$ of vertices of the other fire perimeter. Subsequently, when comparing pairs $(e_m(k_m), n_m(k_m))$ and $(e_s(k_s), n_s(k_s))$, the starting vertex at $k_m = 1$ or $k_s = 1$ of one of the fire perimeters will be re-ordered to obtain the smallest weighted least-squares error between the polygons.

For simplicity of notation, k is used to represent both k_s and k_m after the interpolation because the simulated fire perimeter and the measured fire perimeter have the same number of vertices. In this paper, interpolation of the fire perimeter is done with standard 2D linear interpolation, where interpolated vertices are introduced on the straight lines connecting the original vertices of the closed polygon, and the procedure of linear interpolation is summarized in Algorithm 1.

Algorithm 1 Linear interpolation of wildfire polygon

Input: vertices of the original approximated polygon

Output: Newly constructed vertices of the interpolated polygon

- 1: Calculate the length of each side of the polygon.
 - 2: Calculate the cumulative side length from the starting point.
 - 3: Find locations with equally distributed length along the side of polygon from the starting point.
 - 4: Construct new polygon vertices
-

Similarly, uncertainties of the original vertices can also be interpolated with respect to the cumulative side length from the starting point. Due to the fact that the interpolation is related to the distance from the starting point, it is easy to verify that interpolation from different starting points will lead to different results. This will be considered in the subsequent section when the weighted least squares are calculated. Linear interpolation may lead to a tiny change of the shape of the original wildfire polygon that is negligible compared to the huge burned area of the wildfire polygon. Therefore, the change of the wildfire polygon caused by the linear interpolation is not considered in this paper.

3. Wildfire error quantification

3.1. Weighted least-squares error

With the interpolated (and properly ordered) closed polygons of the simulated fire perimeter $(e_s(k), n_s(k))$, and the measured fire perimeter $(e_m(k), n_m(k))$ with an uncertainty $r(k)$ on each vertex $k = 1, 2, \dots, N$, a weighted least-squares error

$$\frac{1}{N} \sum_{k=1}^N w(k)^2 \left[(e_s(k) - e_m(k))^2 + (n_s(k) - n_m(k))^2 \right], \quad w(k) = \frac{1}{r(k)} \quad (2)$$

can be used to define the distance between the fire perimeters. The interpolated weighting $w(k) = 1/r(k)$ ensures measurements with a large uncertainty $r(k)$ are weighted less in the error characterization. However, even with uncertainty radii defined by (1) with a minimum value r_{min} , the weighted least-squares error in (2) will be skewed and emphasizes parts of the closed-loop polygon where vertices are closely clustered and have only small distances with respect to each other, as also illustrated in Fig. 1. The reasons are clear:

- Small uncertainty radii $r(k)$ due to (1) will result in a large weighting $w(k) = 1/r(k)$ on the regions of the polygon where vertices are closely clustered.
- More vertices in areas of the polygon where vertices are clustered further accentuates the weighting on these regions of the polygon.

To solve the problem of the skewed emphasis of the weighted least-squares error, the weighting $w(k_m)$ for each vertex of the original measured fire perimeter before the interpolation is skew compensated via

$$\tilde{w}_p(k_m) = w(k_m)w_c(k_m)w_u(k_m), \quad w(k_m) = \frac{1}{r(k_m)} \quad (3)$$

where $w_c(k_m)$ is a concentration weighting for each vertex used to account for clustering of vertices on the closed polygon, and $w_u(k_m)$ is a user-defined weighting for each vertex, used to actually emphasize certain vertices on the closed polygon. The weighting $w_c(k_m)$ is defined as

$$w_c(k_m) = \frac{1}{m(k_m)} \quad (4)$$

where $m(k_m)$ is the number of successive vertices around the k_m -th vertex with a small adjacent distance $l(k_m)$ that is defined by the relative distance condition

$$\frac{l(k_m)}{l_{mean}} < 0.2, \quad l_{mean} = \frac{1}{N_m} \sum_{k_m=1}^{N_m} l(k_m)$$

where $l(k_m)$ was defined in (1). The weighting $w_u(k_m)$ is defined to be 0 for the barrier points, defined as the vertices where the fire perimeter has not changed, and 1 for the other vertices.

An illustration of the skew compensation is shown in Fig. 2. On account of the fact that barrier points will not move with the spread of the wildfire, a zero value weighting is assigned to each barrier point. Hence, the weighting radii of barrier points are infinitely large, and not included in Fig. 2.

Finally, to also address the re-ordering of the vertices of the closed polygon, consider the short-hand notation based on complex numbers

$$\begin{aligned} x(k) &= e_s(k) + j \cdot n_s(k), & k &= 1, 2, \dots, N \\ y(k, q) &= e_m(k) + j \cdot n_m(k), & k &= q, q+1, \dots, N, 1, \dots, q-1 \end{aligned} \quad (5)$$

where $x(k) \in \mathbb{C}$ for $k = 1, 2, \dots, N$ represents the 2D coordinates of vertices of a closed polygon of a simulated fire perimeter starting at index $k = 1$, and $y(k, q) \in \mathbb{C}$ represents the 2D coordinates of vertices of a closed polygon of a measured (and possibly interpolated) fire perimeter, but reordered to start at index q . The ability to adjust the

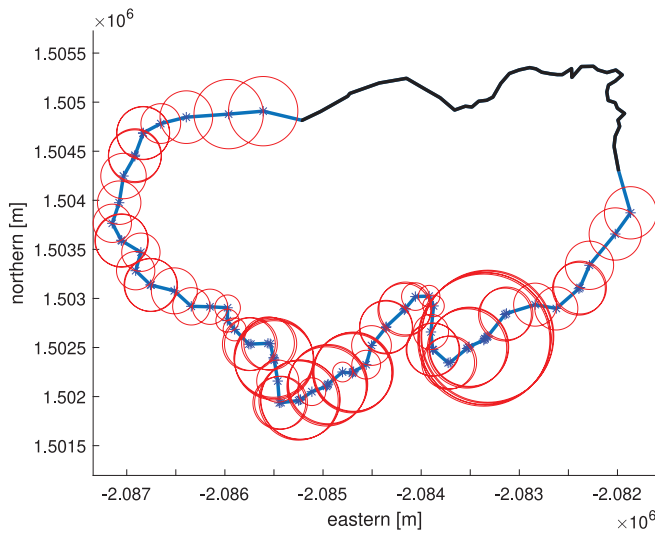


Fig. 2. Weighting radii $1/\tilde{w}_p(k_m)$ (red circles) for skew compensated least-squares compensation on the vertices (blue stars) and barrier points (black line) of a closed polygon of a measured fire perimeter.

starting point $k = q$ of the closed polygon now allows for the definition of the skew compensated weighted least-squares error

$$s_1 = \min_q \frac{1}{N} \sum_{k=1}^N \tilde{w}_p(k)^2 |y(k, q) - x(k)|^2 \quad (6)$$

where $\tilde{w}_p(k)$ is the interpolated $\tilde{w}_p(k_m)$ defined in (3). The starting point $k = q$ is used to remove the dependency of cyclical ordering of complex points describing the closed polygon.

3.2. Weighted surface area error

Surface Area of a Closed Polygon

Consider a 2D polygon of a measured fire perimeter given by the N coordinates of the 2D vertices

$$\begin{bmatrix} e(k) \\ n(k) \end{bmatrix}, \quad k = 0, 1, 2, \dots, N-1$$

ordered by the index k . For the simplicity of the following derivation in this section, k starts from 0 instead of 1, and for an index $k = N$, it is obtained that

$$\begin{bmatrix} e(N) \\ n(N) \end{bmatrix} = \begin{bmatrix} e(0) \\ n(0) \end{bmatrix} \quad (7)$$

formally making the 2D polygon a closed polygon. For such a closed polygon, the total surface area S can be computed by taking a signed sum of the surface area of triangular sub-polygons as follows. Consider a triangular 2D sub-polygon that consists of 3 vertices

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} e(k) \\ n(k) \end{bmatrix} \text{ and } \begin{bmatrix} e(k+1) \\ n(k+1) \end{bmatrix}, \quad k = 0, 1, 2, \dots, N-1 \quad (8)$$

which will have a surface area that can be computed by

$$\frac{1}{2} \left| \begin{bmatrix} e(k) \\ n(k) \end{bmatrix} \times \begin{bmatrix} e(k+1) \\ n(k+1) \end{bmatrix} \right|, \quad k = 0, 1, 2, \dots, N-1 \quad (9)$$

where again the property of a closed polygon in (7) is used in case $k = N-1$. In (9), the symbol $\times$ denotes the cross product and $|\cdot|$ denotes the length of a (cross product) vector. For the 2D vertices, the computation simplifies to

$$\frac{1}{2} |e(k)n(k+1) - n(k)e(k+1)|, \quad k = 0, 1, 2, \dots, N-1$$

by writing out the cross product in terms of the $(e(k), n(k))$ and $(e(k+1), n(k+1))$ coordinates. Let $T(k) = e(k)n(k+1) - n(k)e(k+1)$. According to the shoelace formula, or surveyor's area formula [18], if the polygon is counterclockwise oriented, which means the direction from $(e(k), n(k))$ to $(e(k+1), n(k+1))$ is counterclockwise, then $T(k)$ is positive when the origin point $(0,0)$ is on the left side of the edge (facing towards $(e(k+1), n(k+1))$ from $(e(k), n(k))$). Correspondingly, $T(k)$ is negative when the origin point is on the right side of the edge. Therefore, when the 2D closed polygon is oriented counterclockwise, the area of the polygon S can be expressed by the signed sum of the surface area of triangular sub-polygons as follows:

$$S = \sum_{k=0}^{N-1} \frac{1}{2} (e(k)n(k+1) - n(k)e(k+1)). \quad (10)$$

Expectation and Variance

Now let the subsequent vertices at index k and $k+1$ not be given by a single 2D point, but given by a normal probability distribution

$$\begin{bmatrix} \bar{e}(k) \\ \bar{n}(k) \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} e(k) \\ n(k) \end{bmatrix}, P(k) \right) \text{ and } \begin{bmatrix} \bar{e}(k+1) \\ \bar{n}(k+1) \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} e(k+1) \\ n(k+1) \end{bmatrix}, P(k+1) \right) \quad (11)$$

where $P(k) > 0$ and $P(k+1) > 0$ denote the covariance matrix of the vertices. The covariance matrix is used to model the (joint) probability between the $e(k)$ - and $n(k)$ -coordinates of each vertex. Assume that different vertices are independent with each other and $\bar{e}(k)$ and $\bar{n}(k)$ are uncorrelated (the uncertainty on a vertex is expressed by a circle around each vertex), then

$$P(k) = \begin{bmatrix} \sigma_e^2(k) & 0 \\ 0 & \sigma_n^2(k) \end{bmatrix}, \quad P(k+1) = \begin{bmatrix} \sigma_e^2(k+1) & 0 \\ 0 & \sigma_n^2(k+1) \end{bmatrix} \quad (12)$$

where $\sigma_e(k)$ and $\sigma_n(k)$ are the standard deviations of $\bar{e}(k)$ and $\bar{n}(k)$ respectively. Inspired by [20], the expectation and the variance of $\bar{e}(k)\bar{n}(k+1) - \bar{n}(k)\bar{e}(k+1)$ for each triangular sub-polygon can be calculated based on $T(k)$ as

$$\begin{aligned} \mathbf{E}[\bar{e}(k)\bar{n}(k+1) - \bar{n}(k)\bar{e}(k+1)] &= \mathbf{E}[\bar{e}(k)] \mathbf{E}[\bar{n}(k+1)] - \mathbf{E}[\bar{n}(k)] \mathbf{E}[\bar{e}(k+1)] \\ &= e(k)n(k+1) - n(k)e(k+1). \end{aligned} \quad (13)$$

The variance of $\bar{e}(k)\bar{n}(k+1) - \bar{n}(k)\bar{e}(k+1)$ is

$$\begin{aligned} \text{Var}[\bar{e}(k)\bar{n}(k+1) - \bar{n}(k)\bar{e}(k+1)] &= \mathbf{E}[(\bar{e}(k)\bar{n}(k+1) - \bar{n}(k)\bar{e}(k+1) - \mathbf{E}[\bar{e}(k)\bar{n}(k+1) - \bar{n}(k)\bar{e}(k+1)])^2] \\ &= \mathbf{E}[\bar{e}(k)^2 \bar{n}(k+1)^2] + \mathbf{E}[\bar{n}(k)^2 \bar{e}(k+1)^2] \\ &\quad + \mathbf{E}[e(k)^2 n(k+1)^2] + \mathbf{E}[n(k)^2 e(k+1)^2] \\ &\quad - 2\mathbf{E}[\bar{e}(k)\bar{n}(k+1)\bar{n}(k)\bar{e}(k+1)] - 2\mathbf{E}[\bar{e}(k)\bar{n}(k+1)e(k)n(k+1)] \\ &\quad + 2\mathbf{E}[\bar{e}(k)\bar{n}(k+1)n(k)e(k+1)] + 2\mathbf{E}[\bar{n}(k)\bar{e}(k+1)e(k)n(k+1)] \\ &\quad - 2\mathbf{E}[\bar{n}(k)\bar{e}(k+1)n(k)e(k+1)] - 2\mathbf{E}[e(k)n(k+1)n(k)e(k+1)] \\ &= \mathbf{E}[\bar{e}(k)^2 \bar{n}(k+1)^2] + \mathbf{E}[\bar{n}(k)^2 \bar{e}(k+1)^2] - e(k)^2 n(k+1)^2 - n(k)^2 e(k+1)^2 \\ &= (e(k)^2 + \sigma_e^2(k)) (n(k+1)^2 + \sigma_n^2(k+1)) \\ &\quad + (n(k)^2 + \sigma_n^2(k)) (e(k+1)^2 + \sigma_e^2(k+1)) \\ &\quad - e(k)^2 n(k+1)^2 - n(k)^2 e(k+1)^2. \end{aligned} \quad (14)$$

The expectation and variance of the surface area of the whole closed polygon can then be calculated as

$$\mathbf{E}(S) = \frac{1}{2} \sum_{k=0}^{N-1} (e(k)n(k+1) - n(k)e(k+1)) \quad (15)$$

$$\begin{aligned} \text{Var}(S) = & \sum_{k=0}^{N-1} [(e(k)^2 + \sigma_e^2(k)) (n(k+1)^2 + \sigma_n^2(k+1)) \\ & + (n(k)^2 + \sigma_n^2(k)) (e(k+1)^2 + \sigma_e^2(k+1)) \\ & - e(k)^2 n(k+1)^2 - n(k)^2 e(k+1)^2] \end{aligned} \quad (16)$$

For further simplifying the calculation [17], Eq. (15) can be transformed as follows. By defining $e(N+1) = e(1)$ and $n(N+1) = n(1)$, it can be observed that

$$\begin{aligned} \text{E}(S) = & \frac{1}{2} \sum_{i=0}^{N-1} [e(k)n(k+1) - n(k)e(k+1)] \\ = & \frac{1}{2} \sum_{i=0}^{N-1} e(k)n(k+1) - \frac{1}{2} \sum_{i=0}^{N-1} e(k+1)n(k) \\ = & \frac{1}{2} \sum_{i=1}^N e(k)n(k+1) - \frac{1}{2} \sum_{i=1}^N e(k)n(k-1) \\ = & \frac{1}{2} \sum_{i=1}^N e(k) (n(k+1) - n(k-1)) \end{aligned} \quad (17)$$

For the third equality in (17), $e(0)n(1) = e(N)n(N+1)$ is applied in the first sum and index shifting is used in the second sum.

Weighted Surface Area Error

With the simulated fire perimeter $(e_s(k_s), n_s(k_s))$, $k_s = 0, 1, \dots, N_s - 1$, and the measured fire perimeter $(e_m(k_m), n_m(k_m))$ with an uncertainty mentioned in Section 2.1 on each vertex $k_m = 0, 1, \dots, N_m - 1$. A set of new closed polygons with vertices $(e_d(k_d), n_d(k_d))$, $k_d = 0, 1, \dots, M_d - 1$ can be obtained by finding the union minus the intersection of the simulated fire polygon and the measured fire polygon. Assume that the number of the newly created polygons is L , and the numbers of vertices included in each polygon are M_d , with $d = 1, 2, \dots, L$. The weighted surface area S_d^w of the closed polygon with vertices $(e_d(k_d), n_d(k_d))$, $k_d = 0, 1, \dots, M_d - 1$ can be expressed by

$$S_d^w = \frac{\text{E}(S_d)^\gamma}{\text{Var}(S_d)^{(1-\gamma)}} \quad (18)$$

where γ and $1 - \gamma$ are the weightings added on the expected value and the variance of the surface area respectively, and $\text{E}(S_d)$ and $\text{Var}(S_d)$ can be calculated by (16) and (17) respectively. With the assumption that the uncertainty on a vertex is a circle around the vertex and there are only uncertainties on the measured fire perimeter, it can be achieved that

$$\begin{aligned} \sigma_e(k_d) = \sigma_n(k_d) = & \begin{cases} 0, & \text{if } (e_d(k_d), n_d(k_d)) \neq (e_m(k_m), n_m(k_m)), \\ 1/\tilde{w}_s(k_m), & \text{if } (e_d(k_d), n_d(k_d)) = (e_m(k_m), n_m(k_m)), \end{cases} \\ \tilde{w}_s(k_m) = & w(k_m)w_c(k_m) \end{aligned} \quad (19)$$

where $\sigma_e(k_d)$, $\sigma_n(k_d)$ are the standard deviations defined in (12), and $w(k_m)$, $w_c(k_m)$ are established in (3) with no interpolation. Based on (18), the weighted error of the whole surface area of the union minus the intersection of the simulated fire perimeter and the measured fire perimeter can be defined as

$$s_2 = \sum_{d=1}^L S_d^w = \sum_{d=1}^L \frac{\text{E}(S_d)^\gamma}{\text{Var}(S_d)^{(1-\gamma)}}. \quad (20)$$

With the definition of $\tilde{w}_s(k_m)$ in (19), γ is recommended to be chosen as a value less than or equal to 0.1. Smaller weighting is put on the variance to avoid the erroneous results. For example, if the vertices with extremely large uncertainties are assigned to all the polygons created by the union minus the intersection of the simulated fire polygon and measured fire polygon, then the weighted surface area error is close to zero, and the corresponding simulated fire polygon will be chosen as the optimal simulation that makes no sense. In this paper, γ is picked as 0.1.

4. Wind condition estimation with FARSITE

4.1. Forward simulations

In this study, FARSITE is used for the forward simulation of the simulated fire perimeter $x(k)$ as a function of the single wind vector u . FARSITE can be considered as a non-linear mapping $\rho(\cdot)$ for fire progression, simplified to

$$x(k) = \rho(p(k), u, \theta, \Delta_T) \quad (21)$$

where the input $p(k) \in \mathbb{C}^{N_p}$ is a closed polygon of N_p vertices representing the initial fire perimeter. The simulated output $x(k) \in \mathbb{C}^{N_x}$, defined earlier in (5), is a closed polygon with $k = 1, 2, \dots, N_x$ vertices representing the simulated fire perimeter obtained after a time step of Δ_T . The additional inputs u represents a single wind vector, and θ denotes a parameter representing fuel content, fuel moisture and topography, all assumed to be constant over the time step of Δ_T .

Unknown wind conditions influence the interpolated and re-ordered vertices of the measured fire perimeter represented by the closed polygon $y(k, q)$ defined in (5). The two-dimensional wind vector u in terms of wind speed and wind direction will also influence the vertices of the simulated fire perimeter represented by the closed polygon $x(k)$ and the weighted surface area S_d^w in (18). Along with the definition of the weighting $\tilde{w}_p(k_m)$ in (3), $\tilde{w}_s(k_m)$ in (19), and γ in (18), it is expected that a minimization of s_1 in (6), and s_2 in (20) as a function of u will provide the best wind vector to minimize the distance between $x(k)$ and $y(k)$, and the surface area of the subtraction between the union and the intersection of the simulated and the measured fire polygon, respectively.

4.2. Wind speed and wind direction optimization

The formal problem of finding an estimate of the single wind vector on the basis of a wildfire measurement $y(k)$ using skew compensated weighted least-squares error can be stated as the optimization

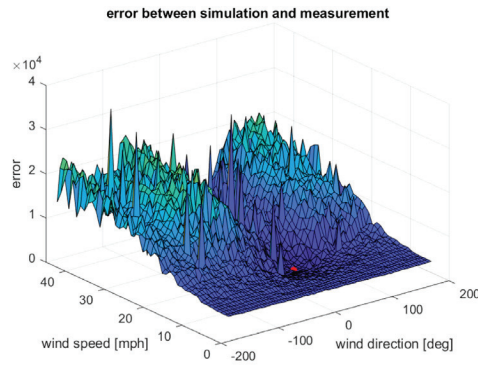
$$\begin{aligned} \min_u s_1(u), \quad s_1(u) = & \min_q \frac{1}{N} \sum_{k=1}^N \tilde{w}_p(k)^2 |y(k, q) - x(k)|^2 \\ x(k) = & \rho(p(k), u, \theta, \Delta_T) \end{aligned} \quad (22)$$

where $\tilde{w}_p(k)$ is the interpolated $\tilde{w}_p(k_m)$ defined in (3) and $y(k, q)$ is defined in (5). Similarly, with weighted surface area error, the formal problem can be stated as the optimization

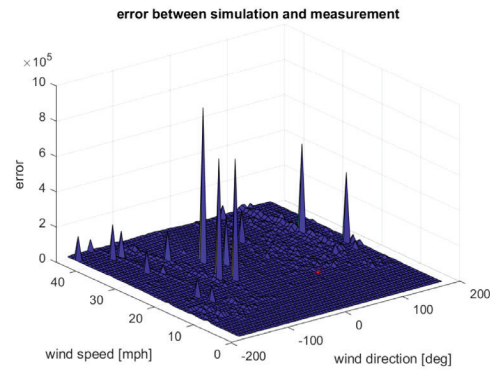
$$\min_u s_2(u), \quad s_2(u) = \sum_{d=1}^L \frac{\text{E}(S_d)^\gamma}{\text{Var}(S_d)^{(1-\gamma)}}. \quad (23)$$

where γ is defined in (18), and $\text{Var}(S_d)$ and $\text{E}(S_d)$ are defined in (16) and (17). Due to the non-linearity and non-convex mapping of $\rho(\cdot)$, a non-linear and iterative optimization is required, typically using the sensitivity or the gradient.

For FARSITE that is responsible for the mapping in (21), the sensitivity or gradient $\frac{\partial}{\partial u} \rho(p(k), u, \theta, \Delta_T)$ is unknown. Numerical evaluation of the gradient is computationally expensive and moreover, FARSITE is known to produce occasional erroneous results at some wind vectors due to numerical problems in interpolation and reconstruction of the main fire perimeter (as will be shown later). These reasons motivate the use of a gradient-free optimization via a grid search and the 2 dimensional size of u motivates a simple 2D gridding procedure over which $s_1(u)$ in (22) and $s_2(u)$ in (23) are evaluated. The 2D grid of u can be updated and refined iteratively to improve the accuracy of the final optimized solution for u . The pseudo-code for the iterative optimization of $s_1(u)$ and $s_2(u)$ are summarized in Algorithm 2. If the skew compensated weighted least-squares error is chosen, let $s(u_{i,j}) =$



(a) Results by using skew compensated weighted least-squares error



(b) Results by using skew compensated weighted surface area error

Fig. 3. Evaluation of skew compensated weighted least-squares error $s_1(u_{i,j})$ and weighted surface area error $s_2(u_{i,j})$ between the simulated and measured fire perimeter at one particular time stamp of the Maria Fire using a 2D grid for the single wind vector $u_{i,j}$. The optimal single wind vector with the lowest value of $s_1(u_{i,j})$ and $s_2(u_{i,j})$ are indicated with a red dot.

$s_1(u_{i,j})$; if the skew compensated weighted surface area error is chosen, let $s(u_{i,j}) = s_2(u_{i,j})$.

Algorithm 2 Optimizing algorithm

Input: θ , $p(k)$, $y(k_m)$, Δ_T , minimum wind condition perturbation λ and stopping criterion ϵ .

Output: Optimized $u \in \mathbb{R}^{2 \times 1}$

- 1: Create n^2 points of a symmetric 2D grid $u_{i,j}$ over a desired range $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, n$ around an initial estimate u_0 of the wind vector.
 - 2: Parallel simulation in FARSITE with $p(k)$, $u_{i,j}$, θ and Δ_T to obtain $x_{i,j}(k)$ for each grid point.
 - 3: Compute the n^2 weighted error $s(u_{i,j})$ over the grid $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, n$, or
 - 4: Find the smallest value $\hat{i}, \hat{j} = \min_{i,j} s(u_{i,j})$ to select the optimized wind vector $u_{\hat{i}, \hat{j}}$.
 - 5: Set $u_0 = u_{\hat{i}, \hat{j}}$ and stop when $|s(u_0 + \lambda) - s(u_0)| \leq \epsilon$ or go back to step 1 to refine grid around u_0 .
-

The skew compensated weighted least-squares error and weighted surface area error are used to determine the difference between the simulated polygon and the measured polygon of wildfire. Simulations can be performed in parallel to speed up the process of finding the optimal wind vector with the above mentioned algorithm.

5. Numerical results

5.1. Maria fire

The Maria Fire ignited in the evening hours of Thursday, October 31, 2019 and consumed well over 4,000 acres (16 km²) within its first several hours of burning. The optimization of the single wind vector is performed for this fire at four different time stamps where measurements of the fire perimeter were available. The objective of the optimization is to improve the fire simulations of the fire perimeters with FARSITE in comparison with the observations obtained at four time stamps.

First we illustrate the results of the gradient-free optimization via a grid search summarized in Algorithm 2 in Fig. 3. The numerical evaluation of the skew compensated weighted least-squares error $s_1(u_{i,j})$ and weighted surface area error $s_2(u_{i,j})$ over a 2D grid $u_{i,j}$ with wind speed from 0 to 45 mph and wind direction from -180 degrees to 175 degrees in Fig. 3 clearly shows the non-differential behavior of $s_1(u)$ and $s_2(u)$,

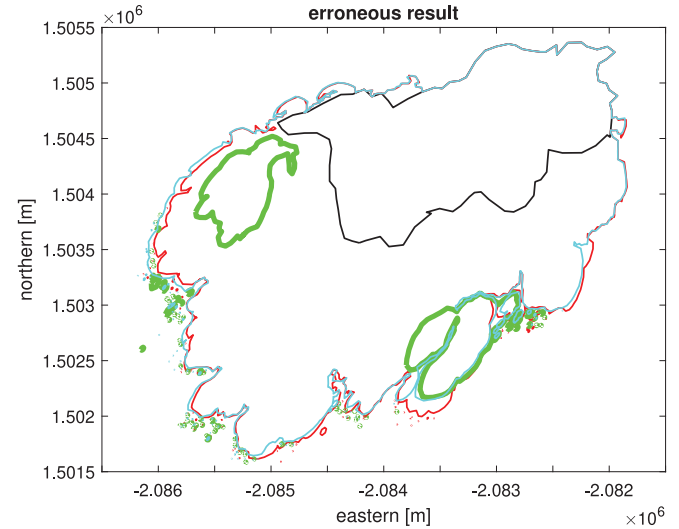


Fig. 4. Simulations of the predicted fire perimeter $x(k)$ with wind speed = 21 mph, wind direction = 34 deg (red), wind speed = 21 mph, wind direction = 35 deg (green), and wind speed = 21 mph, wind direction = 36 deg (cyan), on the basis of the initial fire perimeter $p(k)$ (black).

motivating the use of a gradient-free optimization via a grid search. Sporadic large values of $s(u_{i,j})$ for certain wind vector $u_{i,j}$ are explained by erroneous results due to numerical problems in interpolation and reconstruction of the main fire perimeter by FARSITE, as illustrated in Fig. 4. The simulation results show very similar fire perimeters for two wind vectors that are pretty close to an erroneous result.

Based on gradient-free optimization via a grid search summarized in Algorithm 2, the optimization can correct wildfire simulations when the initial guesses of the wind vectors are not accurate. Correction of the wildfire simulations with two expressions of error for the four different time stamps where measurements of the Maria Fire perimeter were available are summarized in Fig. 5. For each time stamp, the simulated fire perimeter (green lines) based on an initial estimate u_0 of the wind vector obtained from a weather station can be improved (yellow lines) by the optimization of the wind vector via Algorithm 2. It can be observed that the optimized wind vectors provide simulations (yellow lines) that are closer to the measurements (red lines).

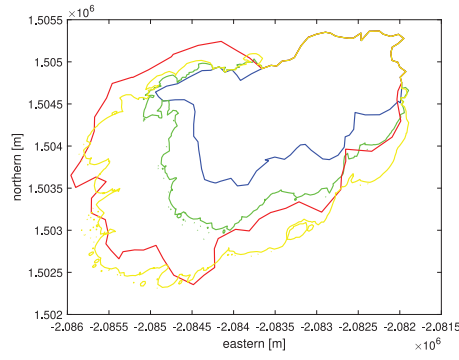
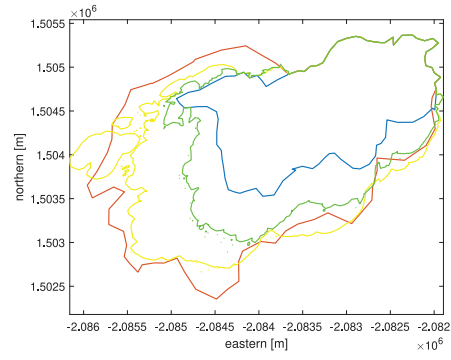
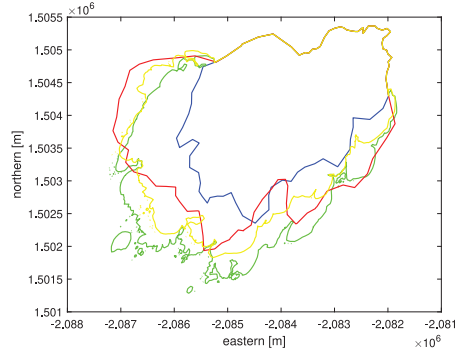
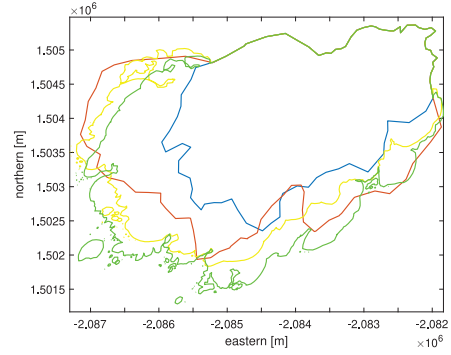
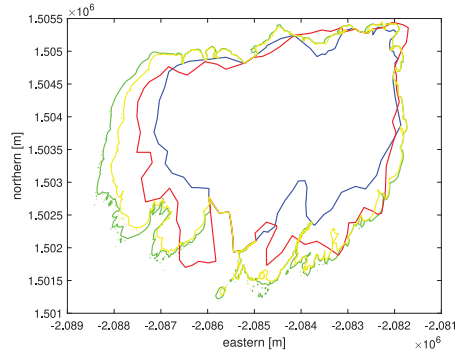
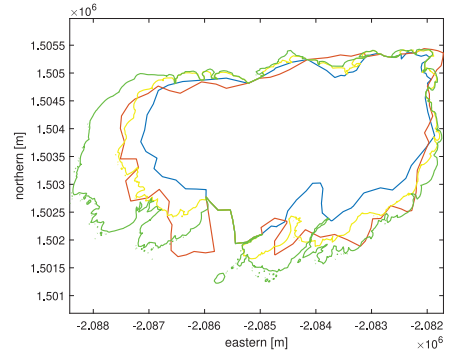
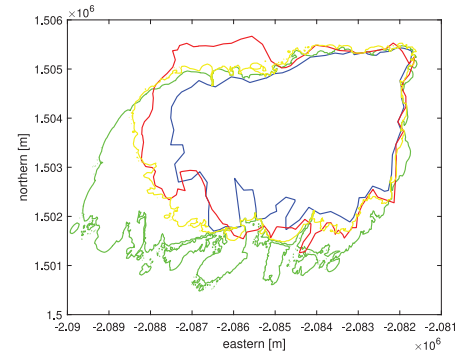
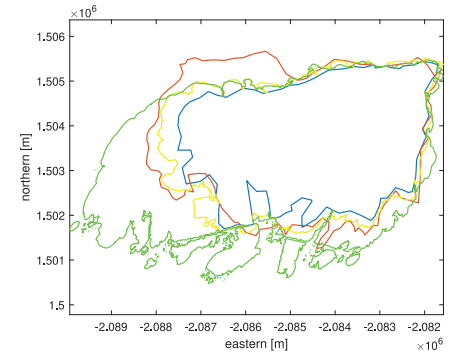
(a) 7:37 p.m. (using $s_1(u)$)(b) 7:37 p.m. (using $s_2(u)$)(c) 7:58 p.m. (using $s_1(u)$)(d) 7:58 p.m. (using $s_2(u)$)(e) 8:31 p.m. (using $s_1(u)$)(f) 8:31 p.m. (using $s_2(u)$)(g) 8:56 p.m. (using $s_1(u)$)(h) 8:56 p.m. (using $s_2(u)$)

Fig. 5. Comparison of the measured and FARSITE simulated fire perimeters for initial u_0 and optimized u wind vector via Algorithm 2 using skew compensated weighted least-squares error $s_1(u)$ and weighted surface area error $s_2(u)$ respectively for the Maria Fire at four different time stamps. Initial ignition (blue); Simulation with initial guess u_0 (green); Simulation with optimized wind vector u (yellow); Measurement at next time stamp (red).

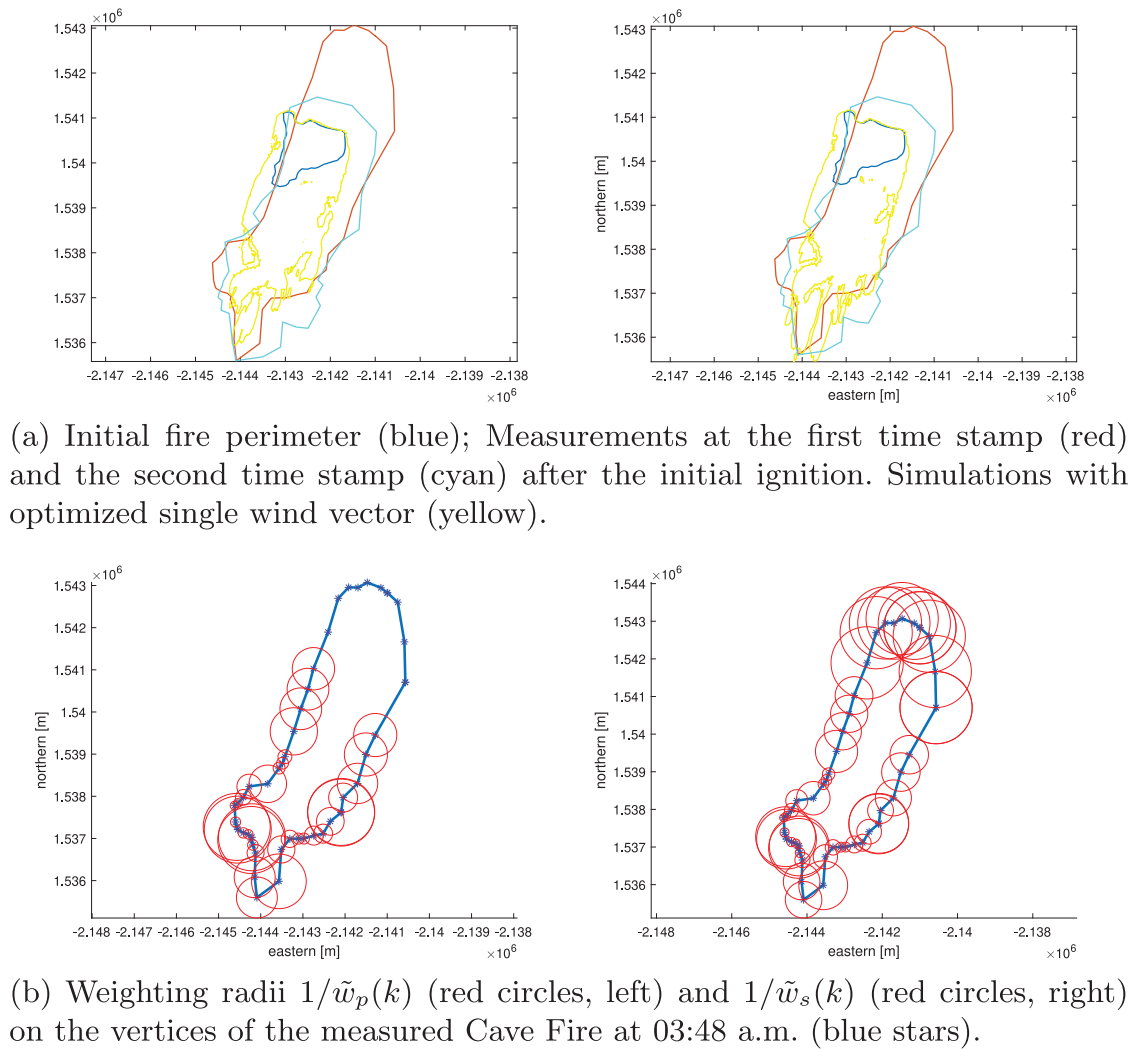


Fig. 6. Simulation and measurements of the Cave Fire with measurement errors in the first measured fire perimeter after the initial fire perimeter. Optimization and weighted radii for skew compensated weighted least-squares error (left), and weighted surface area error (right).

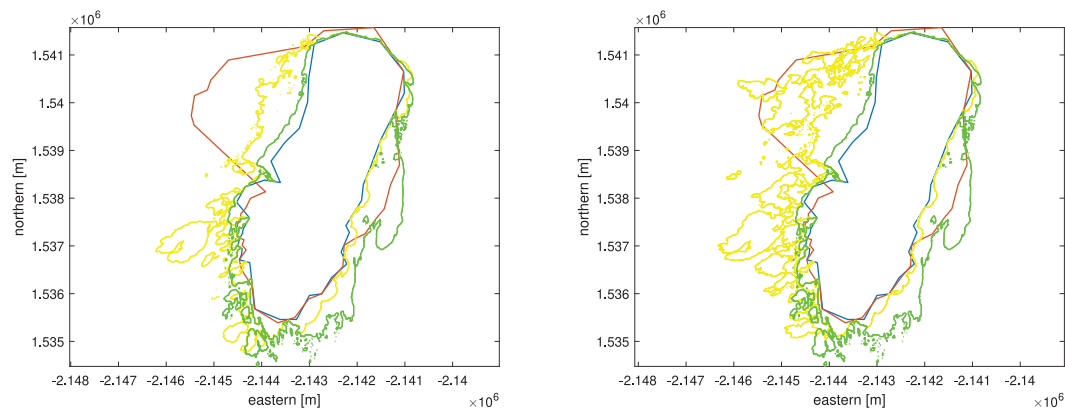
5.2. Cave fire

Although the accuracy of the simulation is improved by using the optimized single wind vector, there are still some parts of the optimized simulation that are somewhat far from the measurement. One reason may be the measurement accuracy, as the combination of aerial surveillance and manually mapped observations is likely to introduce measurement errors. It can also be observed that as the fire perimeter becomes large enough, using only one prevailing wind direction is inadequate for the precise prediction of the wildfire propagation as wind flow is shaped by topography and atmospheric interaction.

The measurement data available for the Cave Fire included here can better demonstrate the two issues of measurement errors and the assumption of a single wind vector. The 2019 Cave Fire started on November 25 and burned more than 3100 acres before being contained. As shown in Fig. 6(a), the top part of the first measurement (after the initial ignition) can be assumed to be wrongly characterized when compared to the second measurement. To be able to account for such errors on the measurement, the weighting $\tilde{w}_p(k_m)$ defined in (3) on the vertices in the top part of the first measurement are adjusted to be zero for the skew compensated weighted least-squares error. The effect of the weighting radii is illustrated in the image on the left in Fig. 6(b). Due to the fact that the measurements in the top part of the first measurement are weighted with 0 for the skew compensated

weighted least-squares error, the corresponding weighting radii are approaching infinity that is not included in the figure. However, the remaining points of the measurement are still allowed to be used for the optimization of the wind vector at this time stamp. For the weighted surface area, on account of the limitation that multiple uncertainties of the vertices will finally act only on one weighted surface area, the infinitely large weighting radii should not be used in case they makes weighted surface area errors of all polygons in the union minus intersection of the simulated fire polygon and measured fire polygon zero. Therefore, the weighting radii of the top part of the measurement are adjusted to be the same as the largest weighting radius in the other parts of the measurement to reflect the large uncertainty in the top part.

When the Cave Fire grows to a large dimension, as illustrated in Fig. 7, it becomes difficult to match the measured fire perimeter with a simulated fire perimeter via single prevailing wind direction. The gradient-free optimization of Algorithm 2 does a better job covering the east side of the fire, but the west side of the fire cannot be accurately covered with a single wind vector due to the topography and atmospheric wind shear effects acting on the fire. This illustrated the limitations of optimizing only a single wind vector.



(a) Results by using skew compensated weighted least-squares error (b) Results by using weighted surface area error

Fig. 7. Comparison of the measurement and FARSITE simulation when the Cave Fire reaches a large dimension. Initial ignition at 05:15 a.m. (blue); Simulation with initial guess of wind vector (green); Simulation with optimized wind vector (yellow); Measurement of fire perimeter (red).

6. Conclusions

This paper shows how fire perimeter measurement can be used to improve the accuracy of a wildfire perimeter simulation, by using the measurement to estimate and correct the prevailing wind speed and wind direction for the simulation. The estimation is based on two carefully defined uncertainty weighted errors. The first error characterization is a skew compensated uncertainty weighted least-squares error that provides a direct comparison of the vertices of a simulated and a measured (noisy) wildfire perimeter. The second error characterization is formulated as a skew compensated uncertainty weighted surface area of the union minus the intersection of a simulated and a measured (noisy) wildfire perimeter. The uncertainty based weighting can account for vertex accuracy and be adjusted for a skewed weighting caused by unequally distributed vertices on the closed polygon of the fire perimeter. In both cases, the (skew compensated) uncertainty radii are used to compute an uncertainty weighted error. A gradient-free optimization via a grid search that uses (refined) grid of the two-dimensional wind vector and exploits parallel computations with FARSITE fire modeling has been done to compute the optimal wind vector. Numerical results on actual wildfire perimeter data obtained from two recent destructive fires in California confirm the improvement of the accuracy of the wildfire perimeter simulations. The skew compensated weighted least-squares error is adept at flexibility of applying complicated uncertainties on vertices, and skew compensated weighted surface area error has an advantage in the simplification of the computational complexity and the reduction of the computational time. Limitations of the proposed methods are due to the optimization of a single wind vector — an assumption that may not hold when a wild fire covers a large area with varying topographical features.

CRedit authorship contribution statement

Li Tan: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft, Visualization. **Raymond A. de Callafon:** Conceptualization, Methodology, Software, Writing – review & editing, Visualization, Supervision. **Jessica Block:** Investigation, Resources, Data Curation. **Daniel Crawl:** Software, Investigation, Resources, Data Curation. **Tolga Çağlar:** Conceptualization. **Ilkay Altıntaş:** Project administration, Funding acquisition.

Declaration of competing interest

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

References

- [1] M.A. Finney, FARSITE, Fire Area Simulator – Model Development and Evaluation, Vol. 4, US Department of Agriculture, Forest Service, Rocky Mountain Research Station, 1998.
- [2] L. Tan, R.A. de Callafon, J. Block, D. Crawl, I. Altıntaş, Improving wildfire simulations by estimation of wildfire wind conditions from fire perimeter measurements, in: International Conference on Computational Science, Springer, 2021, pp. 231–244.
- [3] R.C. Rothermel, A Mathematical Model for Predicting Fire Spread in Wildland Fuels, Vol. 115, Intermountain Forest and Range Experiment Station, Forest Service, United States, 1972.
- [4] N. Cheney, J. Gould, W. Catchpole, The influence of fuel, weather and fire shape variables on fire-spread in grasslands, *Int. J. Wildland Fire* 3 (1) (1993) 31–44.
- [5] J. Mandel, M. Chen, L.P. Franca, C. Johns, A. Puhalskii, J.L. Coen, C.C. Douglas, R. Kremens, A. Vodacek, W. Zhao, A note on dynamic data driven wildfire modeling, in: International Conference on Computational Science, Springer, 2004, pp. 725–731.
- [6] J. Mandel, L.S. Bennethum, M. Chen, J.L. Coen, C.C. Douglas, L.P. Franca, C.J. Johns, M. Kim, A.V. Knyazev, R. Kremens, et al., Towards a dynamic data driven application system for wildfire simulation, in: International Conference on Computational Science, Springer, 2005, pp. 632–639.
- [7] C.C. Douglas, J.D. Beezley, J. Coen, D. Li, W. Li, A.K. Mandel, J. Mandel, G. Qin, A. Vodacek, Demonstrating the validity of a wildfire DDAS, in: International Conference on Computational Science, Springer, 2006, pp. 522–529.
- [8] T. Srivas, T. Artés, R.A. De Callafon, I. Altıntaş, Wildfire spread prediction and assimilation for FARSITE using ensemble Kalman filtering, *Procedia Comput. Sci.* 80 (2016) 897–908.
- [9] T. Srivas, R.A. de Callafon, D. Crawl, I. Altıntaş, Data assimilation of wildfires with fuel adjustment factors in farsite using ensemble kalman filtering, *Procedia Comput. Sci.* 108 (2017) 1572–1581.
- [10] H. Fang, T. Srivas, R.A. de Callafon, M.A. Haile, Ensemble-based simultaneous input and state estimation for nonlinear dynamic systems with application to wildfire data assimilation, *Control Eng. Pract.* 63 (2017) 104–115.
- [11] A. Subramanian, L. Tan, R.A. de Callafon, D. Crawl, I. Altıntaş, Recursive updates of wildfire perimeters using barrier points and ensemble Kalman filtering, in: International Conference on Computational Science, Springer, 2020, pp. 225–236.
- [12] Z. Lin, H.H. Liu, M. Wotton, Kalman filter-based large-scale wildfire monitoring with a system of UAVs, *IEEE Trans. Ind. Electron.* 66 (1) (2018) 606–615.
- [13] Z. Xing, Y. Zhang, C.-Y. Su, Y. Qu, Z. Yu, Kalman filter-based wind estimation for forest fire monitoring with a quadrotor UAV, in: 2019 IEEE Conference on Control Technology and Applications (CCTA), IEEE, 2019, pp. 783–788.
- [14] M.G. Cruz, A.L. Sullivan, J.S. Gould, R.J. Hurley, M.P. Plucinski, Got to burn to learn: the effect of fuel load on grassland fire behaviour and its management implications, *Int. J. Wildland Fire* 27 (11) (2018) 727–741.
- [15] M.G. Cruz, R.J. Hurley, R. Bessell, A.L. Sullivan, Fire behaviour in wheat crops—effect of fuel structure on rate of fire spread, *Int. J. Wildland Fire* 29 (3) (2020) 258–271.
- [16] M.G. Cruz, A.L. Sullivan, J.S. Gould, The effect of fuel bed height in grass fire spread: addressing the findings and recommendations of Moinuddin et al., *Int. J. Wildland Fire* (2018).

- [17] D. Sunday, Fast polygon area and Newell normal computation, *J. Graph. Tools* 7 (2) (2002) 9–13.
- [18] B. Braden, The surveyor's area formula, *College Math. J.* 17 (4) (1986) 326–337.
- [19] C.A. Kolden, P.J. Weisberg, Assessing accuracy of manually-mapped wildfire perimeters in topographically dissected areas, *Fire Ecol.* 3 (2007) 22–31.
- [20] L. Chun, T. Xiaohua, Relationship of uncertainty between polygon segment and line segment for spatial data in GIS, *Geo-Spat. Inf. Sci.* 8 (3) (2005) 183–188.



Li Tan received his B.S. degree in Mechanical Engineering from Shandong University, China, and M.S. degree in Mechanical Engineering from Virginia Tech, USA. He is currently a Ph.D. candidate in Mechanical Engineering at the University of California, San Diego, USA. His current research area is focused on data conditioning and data assimilation for wildfire modeling.



Raymond A. de Callafon received his M.Sc. (1992) and his Ph.D. (1998) in Mechanical Engineering from the Delft University of Technology and is currently a Professor with the Dynamic Systems and Control group in the Dept. of Mechanical and Aerospace Engineering at the University of California, San Diego. He is involved in teaching, research, consulting and judicial expertise in the area of signal processing, parameter/state estimation, servo/adaptive control and embedded software development.

The work of Prof. de Callafon has been applied to structural damage detection problems, adaptive feedback tuning in active noise/vibration control, wildfire data assimilation and dynamic modeling and control of electric power systems.

Prof. de Callafon has received the 2010 Technical Achievement Award from the Information Storage Industry Consortium (INSIC), a 2014 High Performance Computing (HPC) Editors Choice Award for building a cyberinfrastructure to better monitor, predict, and mitigate future wildfires and a 2016 North American Synchrophasor Initiative (NASPI) MVP award.



Jessica Block is the Associate Director for Operational Programs of the WIFIRE Lab at the San Diego Supercomputer Center, UCSD. Her work aims to mitigate disasters using emerging technologies including satellites, sensor networks, machine learning tools, and virtual reality. She has spent her career addressing fire, water, and geologic risks in the American West and Southeastern Australia. In 2019, she was appointed by Gov. Gavin Newsom to the California Wildfire Safety Advisory Board.



Daniel Crawl is the Associate Director for Products of the WIFIRE Lab at the San Diego Supercomputer Center, UCSD. He is one of the lead developers of the Kepler Scientific Workflow System and has advanced experience in working with applications of workflows in bioinformatics, geoinformatics, oceanography, and computational physics. He is the lead architect for the overall integration of many modules in Kepler including distributed data parallel (DDP) execution patterns.



Tolga Caglaris is a data scientist with a background in modeling and simulations of physical and biological phenomena. He has a Ph.D. in statistical physics on phase transitions and critical phenomena. In his postdoctoral years, he worked closely with experimentalists in microbiology for a collaborative interdisciplinary research on finding principles of microbial ecology in the global oceans. He has experience in database methods, data mining and feature engineering, 3D spatial image analysis and molecular dynamics simulations. His contribution to SDSC includes modeling predictive behaviors and increasing the effectivity of the fire simulations for prevention of wildfires.



İlkay Altıntaş is the Chief Data Science Officer and the Founding Director of the Workflows for Data Science (WorDS) Center of Excellence and WIFIRE Lab at the San Diego Supercomputer Center, UCSD. She has worked on different aspects of scientific workflows as a principal investigator and in other leadership roles across a wide range of cross-disciplinary projects. She is a co-initiator of and an active contributor to the open-source Kepler Scientific Workflow System.