

# Interpolation framework to speed up near-surface wind simulations for data-driven wildfire applications

O. Rios<sup>A</sup>, W. Jahn<sup>B</sup>, E. Pastor<sup>A</sup>, M. M. Valero<sup>A</sup> and E. Planas<sup>A,C</sup>

<sup>A</sup>Department of Chemical Engineering, Centre for Technological Risk Studies, Universitat Politècnica de Catalunya · BarcelonaTech, Eduard Maristany, 10-14, E-08019 Barcelona, Catalonia, Spain.

<sup>B</sup>Departamento de Ingeniería Mecánica y Metalúrgica, Pontificia Universidad Católica de Chile, Santiago, Chile.

<sup>C</sup>Corresponding author. Email: eulalia.planas@upc.edu

**Abstract.** Local wind fields that account for topographic interaction are a key element for any wildfire spread simulator. Currently available tools to generate near-surface winds with acceptable accuracy do not meet the tight time constraints required for data-driven applications. This article presents the specific problem of data-driven wildfire spread simulation (with a strategy based on using observed data to improve results), for which wind diagnostic models must be run iteratively during an optimisation loop. An interpolation framework is proposed as a feasible alternative to keep a positive lead time while minimising the loss of accuracy. The proposed methodology was compared with the WindNinja solver in eight different topographic scenarios with multiple resolutions and reference – pre-run – wind map sets. Results showed a major reduction in computation time (~100 times once the reference fields are available) with average deviations of 3% in wind speed and 3° in direction. This indicates that high-resolution wind fields can be interpolated from a finite set of base maps previously computed. Finally, wildfire spread simulations using original and interpolated maps were compared showing minimal deviations in the fire shape evolution. This methodology may have an important effect on data assimilation frameworks and probabilistic risk assessment where high-resolution wind fields must be computed for multiple weather scenarios.

**Additional keywords:** data assimilation, fire behaviour, Rothermel model.

Received 8 February 2017, accepted 13 February 2018, published online 23 April 2018

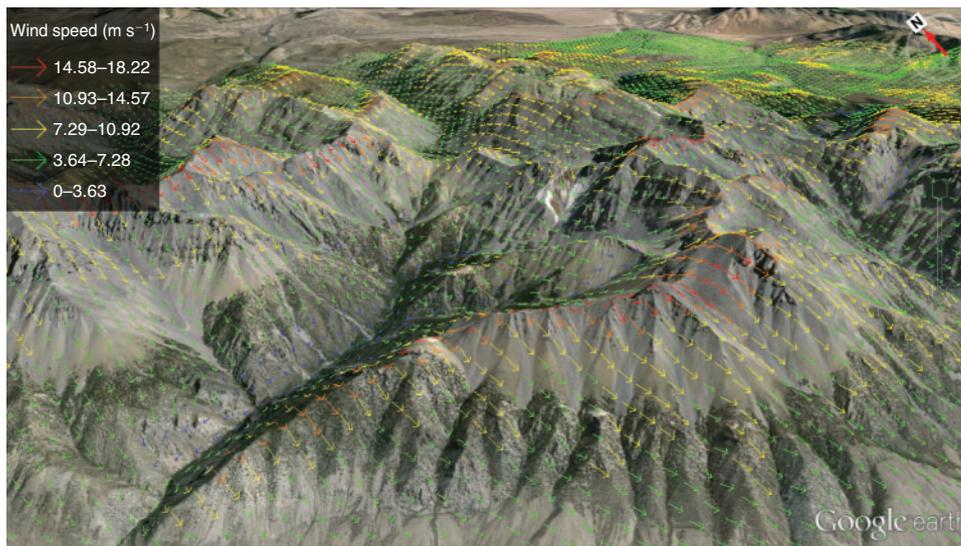
## Introduction

Wind speed and wind direction are two of the most influential parameters for wildfire spread simulations. If wind is present, it is the principal driver for the fire front shape and direction of propagation (Rothermel 1972; Albini 1982). However, the wind speed and direction are not spatially homogeneous along the fire front and they are highly influenced by topography (Forthofer *et al.* 2014b). This fact increases the difficulty of predicting the propagation directions and might cause some extreme behaviour in complex terrain (Viegas and Simeoni 2011; Sharples *et al.* 2012). Thus, it seems necessary to produce accurate wind fields and employ them in fire propagation models in order to deliver detailed fire front forecasts to emergency responders.

Wildfires are known to alter the local weather by inducing strong wind currents that in turn affect the flame structure and ultimately accelerate the fire spread (Clark *et al.* 2004; Filippi *et al.* 2011). Attempting to predict wildfire spread from first principles would thus require a fire-spread model to be coupled to a mesoscale numerical weather prediction model. An example of this approach is WRF-SFIRE (Weather Research and Forecasting model-Spread Fire), which is being used operationally in Israel.

Although this tool has shown promising results, its complexity, required resources (several hundred cores) and initialisation issues mean that it is beyond the current computing capacity of most emergency response services.

An uncoupled strategy to generate high-resolution wind diagnostic models consists of running common numerical weather prediction (NWP) models on smaller grids than their normal application (3 km or larger horizontal grid resolution). Although this prognostic approach has given positive results with resolutions of less than 1 km (Seaman *et al.* 2012; Ching *et al.* 2014), when used in complex terrain, they encounter problems with the computational grids necessary for finite difference discretisation (Lundquist *et al.* 2010) and the turbulence closure models (Wyngaard 2004). This, together with the costly running time, render them unappropriated for most wildfire scenarios. An alternative approach is to exploit mesoscale weather predictions or localised wind field measurements and apply a diagnostic model to downscale the wind maps to high spatial resolutions (up to 15 m) at a low height (below 2 m). The advantages of this strategy are the reduction in computational time (owing to simplified models and the avoidance of time stepping) and the



**Fig. 1.** WindNinja output at 1 m height and grass canopy for a given hilly scenario. Wind vector map representation (direction and magnitude) in Google Earth for  $10 \text{ m s}^{-1}$  and  $0^\circ$  (north) principal wind speed and direction.

accurate integration of digital elevation maps (DEM) with higher resolution than the prognostic approach.

High-resolution diagnostic wind models can be classified into three categories according to the level of physics considered and the strategy followed to resolve it. The diagnostic models of the first type are called mass-conserving models and use fundamental mass-balance equations often combined with empirical correlations for non-neutral vertical stability or diurnal heating dynamics (Forthofer *et al.* 2009; Butler *et al.* 2015). The models within the second category are based on the numerical approximate solution of Navier–Stokes equations together with a turbulence submodel for closure. They can also incorporate energy balance equations. These computational fluid dynamics (CFD) models can deliver a high degree of accuracy but usually require large computational resources. Examples include WindStation (Lopes 2003) –specifically aimed at assisting wildfire simulations – and the more general OpenFOAM (Weller and Tabor 1998). The third category falls in between the two previous categories, as momentum is solved by linearising the conservation equations. Their performance in terms of computing time is similar to mass-conserving models, but despite including more physics they tend to produce less accurate results than mass-conserving models, as demonstrated in scenarios with dispersion of hazardous materials (Homicz 2002).

WindNinja (Forthofer 2007; Forthofer *et al.* 2014b; Wagenbrenner *et al.* 2016) is an open-source software developed and maintained by the USDA Forest Service. It takes into account the terrain (DEM) and three types of canopy cover (grass, brush and trees), and it can use either a mass-conserving or a mass-and-momentum model. The mass-conserving model estimates the result of the mass balance equation, whereas the mass-and-momentum model uses the OpenFOAM toolkit (Weller and Tabor 1998). Although the use of the second approach gives more accurate results for strong winds on the lee side of mountains and ridges (where eddies can occur), the first option

is 60 times faster to deliver the wind map (Forthofer *et al.* 2014a). Additionally, the mass-conserving approach allows the use of point measurements and is able to handle non-neutral atmospheric stability effects. Moreover, its performance might be improved by using mesoscale forecast data for initialisation. This combination can thus account for both mesoscale data and local terrain effects (Forthofer *et al.* 2014b).

WindNinja is one of the most extensively used diagnostic wind models used to generate high-resolution near-surface wind maps for the use in fire propagation models. WindNinja is the core near-surface wind simulator for recognised fire spread simulators such as FARSITE (Finney 1998), FireStation (Lopes *et al.* 2002), FlamMap (Finney 2006) and Wildfire-Analyst (Monedero *et al.* 2011). Fig. 1 shows typical output of WindNinja mapped onto a terrain rendered in Google Earth. Despite the software's reasonably high speed, computation times required even by the fast mass-balance solver are of the order of 5 min for an  $800 \times 800$  cell map using multicore workstations (Sanjuan *et al.* 2016a). Computing times of such magnitude are unaffordable if the wind field is to be computed recursively. This is a critical feature of any data assimilation algorithm incorporated into data-driven simulators. Those simulators (see examples in Altintas *et al.* 2015; Rios *et al.* 2016; Zhang *et al.* 2017) try to match observed fire locations with simulation results. To achieve this, they need to run the fire spread algorithm (which includes a surface wind module) multiple times. Thus, resolving the wind field for different values of mesoscale wind is a major bottleneck when aiming to achieve positive lead times.

In recent years several attempts have been made to develop data-driven wildfire propagation systems. They have been proven to be a promising strategy to overcome the lack of information (such as fire front location, accurate vegetation maps, weather conditions, etc.) inherent in emergency operations, and to reduce the high uncertainty of initial model

parameters (Mandel *et al.* 2009; Rios and Rein 2014; Rochoux *et al.* 2014; Altintas *et al.* 2015; Rios *et al.* 2016). Although different strategies are being investigated, they all require multiple runs of the core spread model, which consists of a wildfire spread algorithm and normally includes the computation of near-surface wind diagnostic models. Therefore, computation time has become a principal constraint if the whole system is to deliver useful information to end users in due time. Recent efforts to parallelise WindNinja exploit domain decomposition methods (Sanjuan *et al.* 2016a), computational parallelisation based on GPU (graphics processing unit) (Sanjuan *et al.* 2016c) and hybrid integration using message passing interphase (MPI) and open multi-processing (OpenMP) (Sanjuan *et al.* 2016b). Despite the remarkable speed-up achieved with those strategies, the computing time required for a single run of a  $1500 \times 1500$  cell map exceeds 90 s on a 64 node processor using the most efficient approach (hybrid MPI-OpenMP integration). WindNinja is thus still not sufficiently fast to be used in an optimisation framework where hundreds of runs are required.

In order to minimise the computational cost, we propose to downscale only a few combinations of wind speed and direction (gridded wind output from WindNinja) and then interpolate the wind field at 1 m above ground level (the lower limit of the downscaling height suitable for this methodology). The interpolation is performed using a set of base fields generated with WindNinja for different wind speeds and directions. This scheme allows to produce wind maps that take into account terrain characteristics while avoiding to run WindNinja iteratively. This approach has an intrinsic limitation when resolving thermally driven winds that might be dominant in complex terrain during quiescent synoptic conditions. However, the approach is valid for any non-quiescent situations where mechanical effects of the terrain dominate over thermal effects. Those are the cases of principal interest for fire applications, as they also yield the highest fire front rate of spread.

In this article, we perform an in-depth analysis of this strategy using the WindNinja mass-conserving solver without thermal parameterisations. The effects of grid resolution, canopy cover and the number of available base maps are studied in order to find the best interpolation framework. The proposed methodology is assessed in different scenarios (illustrating diverse topography configurations) under different conditions by comparing the resulting wind fields with the results obtained using WindNinja directly. Finally, a comparative study is presented to evaluate the effect of the proposed wind field interpolation methodology in a fire spread simulation.

## Interpolation framework

A possible approach to speed up multiple WindNinja runs is to exploit the fact that, within a data-driven application strategy, diagnostic wind models need to be run multiple times for the same scenario (e.g. same DEM, canopy cover, height) changing only the boundary conditions such as the initialisation wind speed and direction (hereafter referred as the principal wind speed and direction), canopy cover, etc. An interpolation framework can be set as follows:

A WindNinja generated wind map ( $\mathbf{W}$ ) consists of two matrices that contain wind speeds ( $\mathbf{U}$ ) and wind directions ( $\mathbf{D}$ )

(in angles) at a requested height ( $h$ ). A wind map  $\mathbf{W}$  is produced by WindNinja from a given representative wind speed, called principal wind speed ( $U$ ) and principal wind direction ( $D$ ) at a certain height ( $h$ ), for a given digital elevation model ( $T$ ), and a fuel type ( $f$ ) that can be either grass, bush or trees.

A set of base map ( $\{\mathbf{W}_b\}$ ), are those wind maps that are used as basis for the interpolation framework. Thus, they are composed by base wind speed ( $\mathbf{U}_b$ ) and base wind directions ( $\mathbf{D}_b$ ). As before, those maps are generated with principal wind speeds ( $U_b$ ) and directions ( $D_b$ ). The subscript ‘ $b$ ’ is to recall that they are principal wind components that define the correspondent base map.

For a singular principal wind speed and an  $n$ -direction WindNinja produces a set of wind maps  $\{\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_n\}$  that can be employed as base wind maps (namely base maps,  $\mathbf{U}_{b,i}, \mathbf{D}_{b,i}$ ). Then, instead of running WindNinja again for a new desired principal wind speed and direction, the wind map ( $\mathbf{U}_I, \mathbf{D}_I$ ) is obtained by interpolating the base maps to the desired principal wind speed ( $U$ ) and direction ( $D$ ) according to the following equations:

$$\mathbf{U}_I = \sum_i^n \mathbf{U}_{b,i}(U_{b,i}, D_{b,i}, f, h, T) \cdot F_{U,i}(U_{b,i}, D_{b,i}, f, h, T, D, U) \quad (1)$$

$$\mathbf{D}_I = \sum_i^n \mathbf{D}_{b,i}(U_{b,i}, D_{b,i}, f, h, T) \cdot F_{D,i}(U_{b,i}, D_{b,i}, f, h, T, D, U). \quad (2)$$

The wind speed proportionality factor ( $F_{U,i}$ ) could in principle depend on the fuel type ( $f$ ), the original principal wind speed ( $U_b$ ). Similar dependencies might be found for the wind direction proportionality factor ( $F_{D,i}$ ). Although the proportionality factor can also depend on the DEM, the hypothesis is that this dependency is negligible. To assess whether such an interpolation scheme exists and whether it has a coherent form for the different dependencies, 11 scenarios were defined and explored. The scenario descriptions are given in Table 1. For each scenario all wind speeds, directions and canopy covers were combined to explore the sensitivity to the parameters.

In total, 1320 WindNinja runs were launched and their output compared. The work flow is represented in Fig. 2 together with the nomenclature used in following sections. To estimate the factors  $F_{D,i}$  and  $F_{U,i}$  (Eqns 1, 2), maps generated at a height of 1 m were compared with maps where both  $U, D$  and  $U_b, D_b$  were changed. Seeing as the values of  $F_D$  and  $F_U$  might differ at each pixel point in the map, mean values and standard deviations were computed.

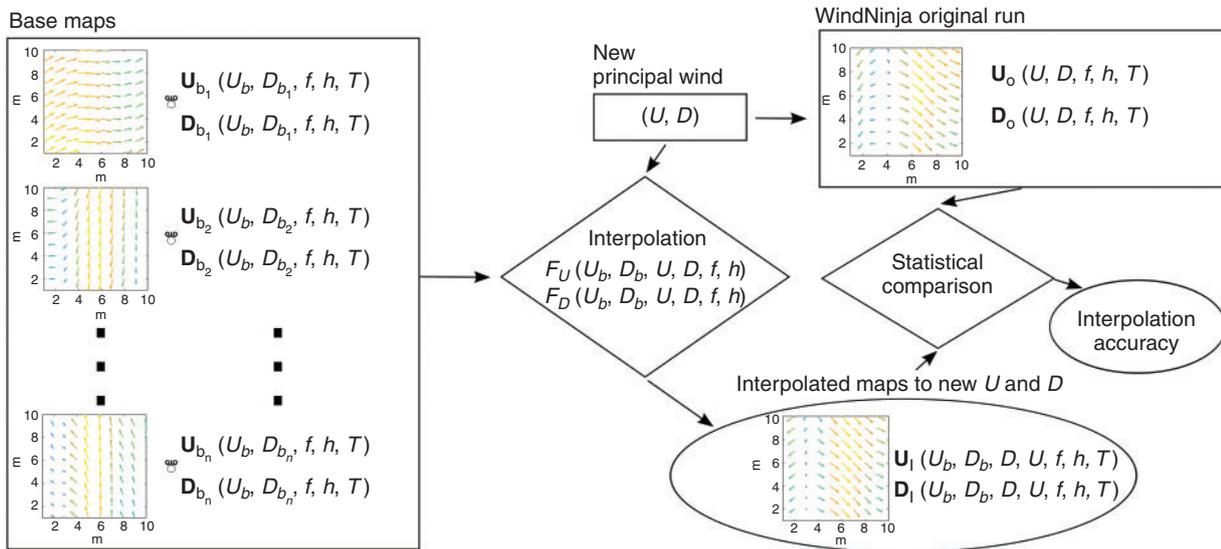
### Principal wind speed dependency

The dependence of the factors ( $F_D$  and  $F_U$ ) on the speed ( $U$ ) and the input base speed ( $U_b$ ) is analysed by comparing speed and direction maps generated with a constant principal wind direction  $D_b$ .  $F_U$  shows a strong linear dependence on  $U$  in all simulated scenarios. However,  $F_D$  shows no clear dependence on  $U$ , as the mean value is very close to unity (no dependence at all). The dispersion of values for averaged speed lower than  $3 \text{ m s}^{-1}$  does increase. Fig. 3 shows the mean values of  $F_U$  (solid lines) and their standard deviation (shadowed areas) for an illustrating scenario, (Alaska30m) as a function of 15 different wind speeds ( $U$ ) and eight base wind speeds ( $U_b$ ).

**Table 1. Simulated scenarios used for model testing**

Abbreviations: IGCC, Institut Geogràfic i Cartogràfic de Catalunya; USGS, Unites States Geological Survey; DEM, digital elevation maps; Res, resolution. Canopy type abbreviations: g, grass; b, brush; t, trees.  $U_b$  and  $D_b$  values are expressed as lower bound (lb), step (s), upper bound (up) and the resulting number of elements ( $n$ ) as: lb:s:up ( $n$ )

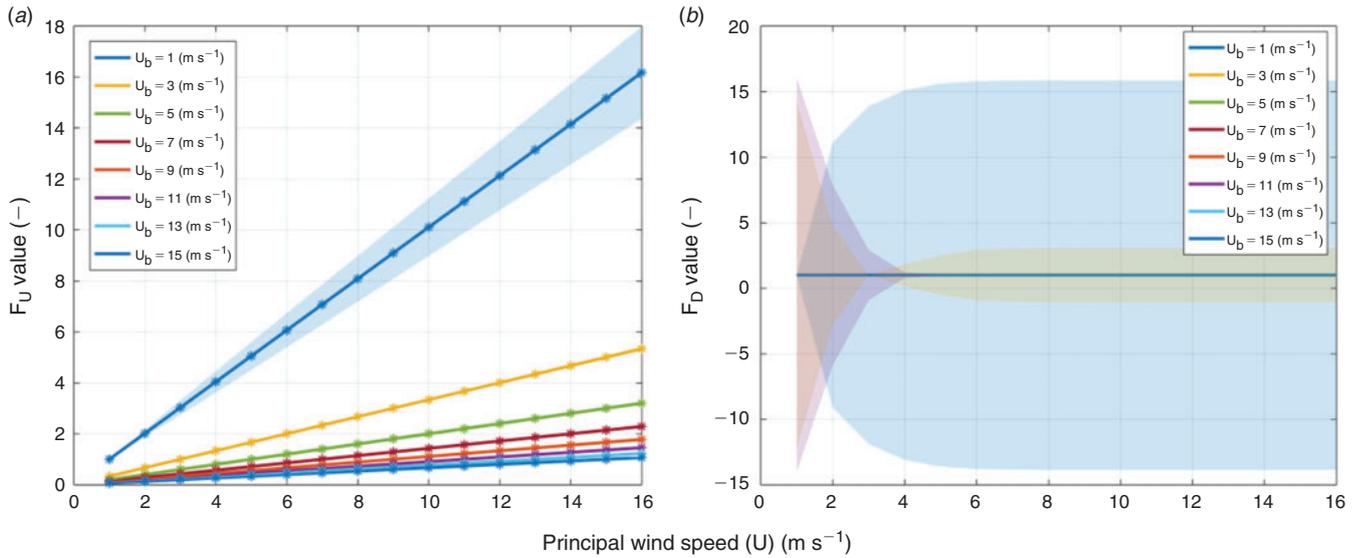
ID	DEM Res (m)	Source	Size (km)	Size (cells × cells)	Canopy	$U_b$ ( $m s^{-1}$ ) ( $n$ )	$D_b$ ° ( $n$ )		
Montseny05m	5	IGCC	4.70	3.65	940	730	g, b, t	1:1:2 (2)	0:15:285 (8)
Montseny15m	15	IGCC	6.81	5.46	454	364	g, b, t	1:2:15 (8)	0:45:315 (8)
Alaska30m	30	USGS	28.20	21.39	940	713	g, b, t	1:2:15 (8)	0:45:315 (8)
Boulder30m	30	USGS	7.20	5.13	240	171	g	5:6:11 (2)	0:5:355 (32)
Canada30m	30	USGS	15.60	10.74	520	358	g, b, t	1:2:15 (8)	0:45:315 (8)
Colorado30m	30	USGS	7.20	5.13	240	171	g	1:2:15 (8)	0:15:345 (25)
Idaho30m	30	USGS	28.20	21.39	940	713	g, b, t	1:1:16 (16)	0 (1)
Missoula30m	30	USGS	6.42	3.78	214	126	g, b, t	1:2:15 (8)	0:45:315 (8)
Maipo90m	90	USGS	19.17	11.88	213	132	g	1:2:15 (8)	0:45:315 (8)
Santiago90m	90	USGS	18.63	12.06	207	134	g	1:2:15 (8)	0:45:315 (8)
Valpo90m	90	USGS	18.81	12.87	209	143	g	1:2:15 (8)	0:45:315 (8)
Montseny05m	5	IGCC	4.70	3.65	940	730	g, b, t	1:1:2 (2)	0:15:285 (8)
Montseny15m	15	IGCC	6.81	5.46	454	364	g, b, t	1:2:15 (8)	0:45:315 (8)
Alaska30m	30	USGS	28.20	21.39	940	713	g, b, t	1:2:15 (8)	0:45:315 (8)
Boulder30m	30	USGS	7.20	5.13	240	171	g	5:6:11 (2)	0:5:355 (32)
Canada30m	30	USGS	15.60	10.74	520	358	g, b, t	1:2:15 (8)	0:45:315 (8)
Colorado30m	30	USGS	7.20	5.13	240	171	g	1:2:15 (8)	0:15:345 (25)
Idaho30m	30	USGS	28.20	21.39	940	713	g, b, t	1:1:16 (16)	0 (1)
Missoula30m	30	USGS	6.42	3.78	214	126	g, b, t	1:2:15 (8)	0:45:315 (8)
Maipo90m	90	USGS	19.17	11.88	213	132	g	1:2:15 (8)	0:45:315 (8)
Santiago90m	90	USGS	18.63	12.06	207	134	g	1:2:15 (8)	0:45:315 (8)
Valpo90m	90	USGS	18.81	12.87	209	143	g	1:2:15 (8)	0:45:315 (8)



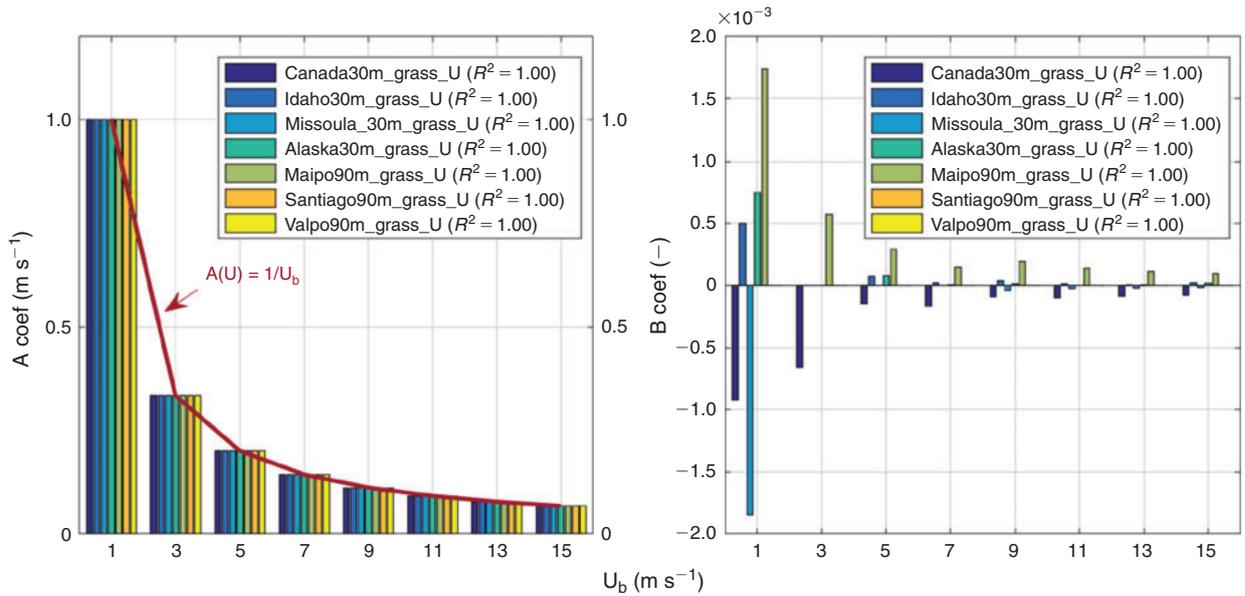
**Fig. 2.** Work flow diagram. Squared boxes are input, diamonds are computational process and oval are outputs. Illustrative wind field maps of a  $10 \times 10$  cells scenario are displayed.

The linear dependency between  $F_U$  and  $U$  (the slopes of the different curves in Fig. 3a) changes as a function of the principal wind ( $U_b$ ) used to create the base map ( $U_b$ ). To quantify this slope, a linear regression of the form  $F_U = AU + B$  was adjusted for every principal wind speed and every testing scenario in

Table 1. Note that the intercept is added for consistency although a value of zero is expected. The results for grass canopy are depicted in Fig. 4. All scenarios show exactly the same slope behaviour for different base wind speeds. This behaviour actually corresponds to the inverse of the base wind speed itself.



**Fig. 3.**  $F_U$  (a) and  $F_D$  (b) interpolating wind factors for different principal wind speed ( $U_b$ ) on Alaska30m. Median values along the domain are depicted as solid lines and standard deviations as shaded areas. The principal wind direction ( $D_b$ ) is set constant to 0°. For the sake of clarity, only four standard deviations (i. e. shaded areas) are depicted in (b). Those are  $U_b = 1$  (light blue), 3 (yellow), 9 m s<sup>-1</sup> (orange) and 11 m s<sup>-1</sup> (purple). Note  $F_D$  mean values for all  $U_b$  speeds overlap at unity.



**Fig. 4.**  $F_U$  linear coefficients dependency on wind speed and grass canopy cover. The slope (left plot) evolution is equivalent for every scenario and corresponds to the slope  $1/U_b$ . In contrast, the origin coefficient (right plot) has a negligible dependency and can be assumed to be zero. The  $R^2$  value is displayed in the legend for each scenario.

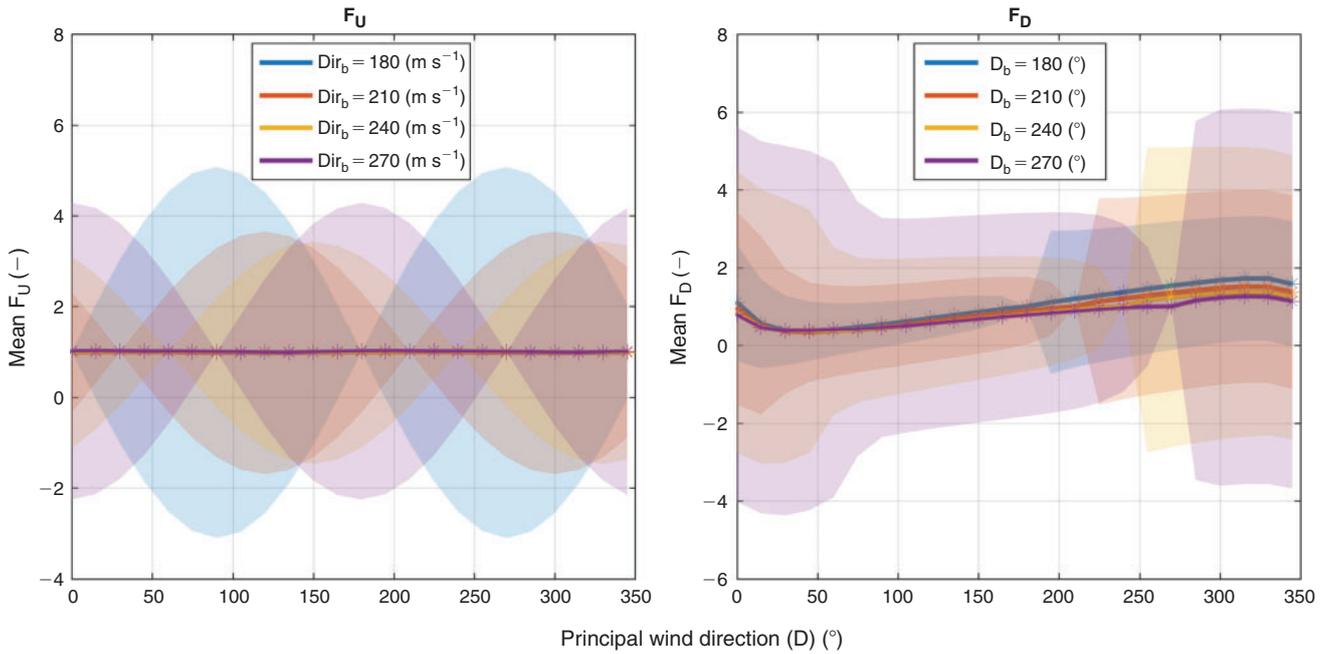
The intercept does show some dependence on the scenario, but as it is of the order of  $\times 10^{-3}$ , it can be neglected (as anticipated). The linear dependence on  $U$  when the direction is fixed could be expected because mass-consistent models simply solve the Poisson equation and Fig. 3 can thus be regarded as a verification result.

The wind speed interpolating factor  $F_U$  can thus be generated using the base map with the closest direction available according to the following relation,

$$F_U(U_b, D_b, U, D) = \frac{1}{U_b} \cdot U|_{D_b} \quad (3)$$

#### Principal wind direction dependency

An analogous analysis is conducted for the dependence of  $F_U$  and  $F_D$  on the wind direction. Results for the Alaska30m case with a base wind speed of 15 m s<sup>-1</sup> are presented in Fig. 5 as an illustration (other scenarios, again, show similar behaviour).



**Fig. 5.** Alaska30m downscaling wind factors dependency on wind direction at 1 m height and with grass canopy cover. Four different principal wind directions ( $D_b$ ) are displayed. Wind principal speed is set to ( $U_b$ )  $15 \text{ m s}^{-1}$ . Shaded area represents the associated s.d.

The speed downscaling factor  $F_U$  shows an averaged value of unity (i.e. no significant influence) when the wind direction changes. The standard deviation, however, shows large values as the angle between the input and the base wind direction increases, reaching a maximum when they are perpendicular to each other (shaded area in Fig. 5). That is, if the base map plot is generated with a base wind direction  $D_b = 0^\circ$ , the discrepancy will be at its maximum with an input wind direction ( $D$ ) of  $90^\circ$ . The standard deviation can reach values five times larger than the absolute mean downscaling value (Fig. 5a). The standard deviation rapidly decreases as the interpolating direction becomes closer to reference direction maps and for symmetrical base wind directions (i.e.  $0\text{--}180, 90\text{--}270$ ). This symmetry implies that the wind speed’s directional dependence is cancelled out for opposite directions. This indicates that wind direction must not differ significantly from the base map in order to get an acceptable downscaling result. This effect increases as the base wind speed gets higher. The base wind direction factor  $F_D$  shows a linear dependence on wind direction for a reasonably large range of averaged wind directions ( $D$ ), but the standard deviation (shaded areas) is larger when changing the averaged wind direction ( $D$ ) than when changing the wind speed (previous case, Fig. 4). The directional dependence of  $F_D$  on the updated direction ( $D$ ) has a large standard deviation when comparing maps that are only some degrees off the base direction (see Fig. 5b). The comparison shows different behaviours for directions to both sides of the base map direction. Four different base reference directions are displayed in Fig. 5b.

Fig. 5 led to the conclusion that more than one base wind direction map is required in order to keep the interpolating error in acceptable bounds because  $F_D$  does not show a clear pattern that would allow for a one-point correction. The proposed framework thus requires a set of pre-run wind directions so that

the wind direction factor  $F_D$  can be defined as a direct linear interpolation between the closest two maps. Different direction sets are investigated in the following sections. The interpolation is formulated as:

$$F_U(U_b, D_b, U, D) = D(U_b, D_{b_k}) + \frac{D(U_b, D_{b_k}) - D(U_b, D_{b_{k+1}})}{U_b - D_{b_{k+1}}} \cdot (D - D_b) \quad (4)$$

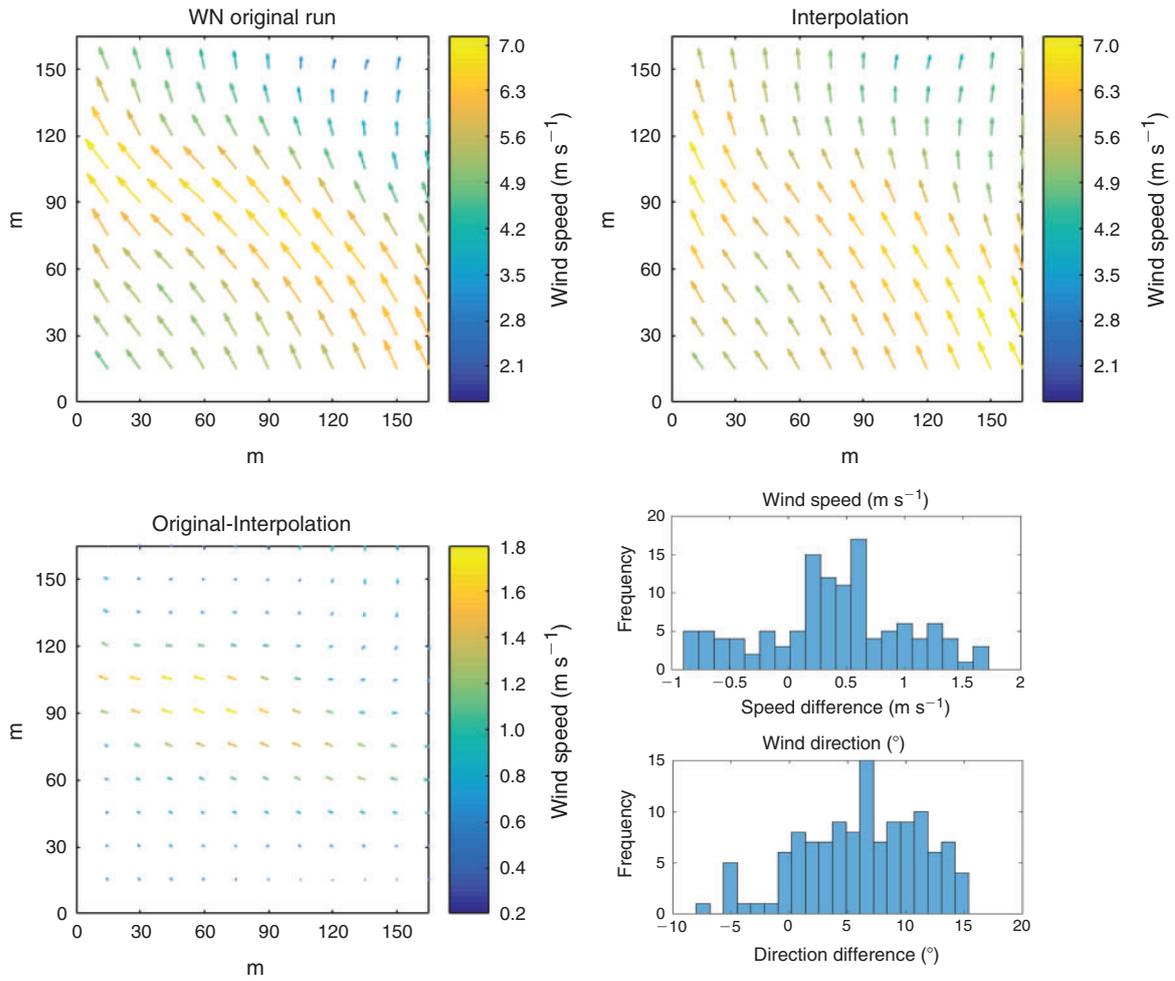
where the subscript  $k$  represents the closest direction map for a given interpolating direction ( $D$ ).

It is worth noting that, because the wind direction is a cyclic, dimensionless magnitude, it must be properly scaled and bounded in order for Eqn 4 to be applied. This is, all angles are devalued to  $[0, 2\pi]$  and all subtraction operations bounded between  $[-\pi, \pi]$  in order to preserve linearity.

Out of the three canopy covers explored (grass, trees and brush), only the last one showed a noticeable difference. Whereas grass and trees exhibit similar output wind fields (the comparison gives an absolute discrepancy distribution centred at 0) the brush canopy has a slowing effect and scatters the directions. Therefore, it is important to use a reference map specially generated for the required canopy, especially if it is the brush canopy.

### Validation methodology

To validate the overall interpolating framework we explore the absolute error when comparing the interpolated wind maps with directly generated WindNinja simulations. The validation is performed step by step. First, only the wind speed is evaluated; second, only the directional correction; lastly, both direction and speed are corrected analysing the effect of the DEM’s mesh resolution. For assessing the discrepancy, the error (E) and the



**Fig. 6.** Composite illustrating the validation process for a Montseny 15m scenario. Only a subset of  $11 \times 11$  cells is shown for clarity. The WindNinja original field ( $U = 9 \text{ m s}^{-1}$  and  $D = 105^\circ$ ) is compared with an interpolated field (from  $U_b = 5 \text{ m s}^{-1}$  and  $D_b = \{90, 180\}$ ). The subtraction result is then statistically studied, in this case in terms of histogram plots for each magnitude.

absolute error (AE) are defined together with their percentages ( $PE$ ,  $APE$ ) as:

$$E_i^U = U_I(T, f, U, D)_i - U_0(T, f, U, D)_i \quad (5)$$

$$AE_i^U = |U_I(T, f, U, D)_i - U_0(T, f, U, D)_i| \quad (6)$$

$$PE_i^U = \frac{E_i^U}{U_0(T, f, U, D)_i} \cdot 100 \quad (7)$$

$$APE_i^U = \frac{AE_i^U}{U_0(T, f, U, D)_i} \cdot 100 \quad (8)$$

where  $|x|$  denotes the absolute value of  $x$ , subscript  $i$  represents a particular pixel in the wind speed matrix, and  $U_0$  and  $U_1$  are the original WindNinja simulated and interpolated wind speed maps respectively. The same definition applies to  $D_0$  (direction maps), although percentage errors in this case are meaningless. When dealing with averaged values of  $AE_i$  and  $APE_i$  metrics,

the abbreviations MAE (mean absolute error) and MAPE (mean absolute percentage error) are used. Fig. 6 displays the validation process for an illustrative subset of  $11 \times 11$  cells. The difference between the original and interpolated wind field (calculated by vector subtraction) is displayed as a histogram for both speed and directions.

The eight different scenarios used for validating the interpolation framework are described in Table 2. Each scenario was simulated with several different DEM resolutions to assess the influence of this parameter on the interpolation. The combination of wind speed and direction (columns 8 and 9) produced 7776 scenarios that were run with WindNinja.

## Results and discussion

### Framework validation

#### Wind speed interpolation

First, the framework presented above was applied to evaluate the downscaling interpolation when the direction of the principal wind used to generate the reference base map matches the

**Table 2. Simulated scenarios used for model validation**

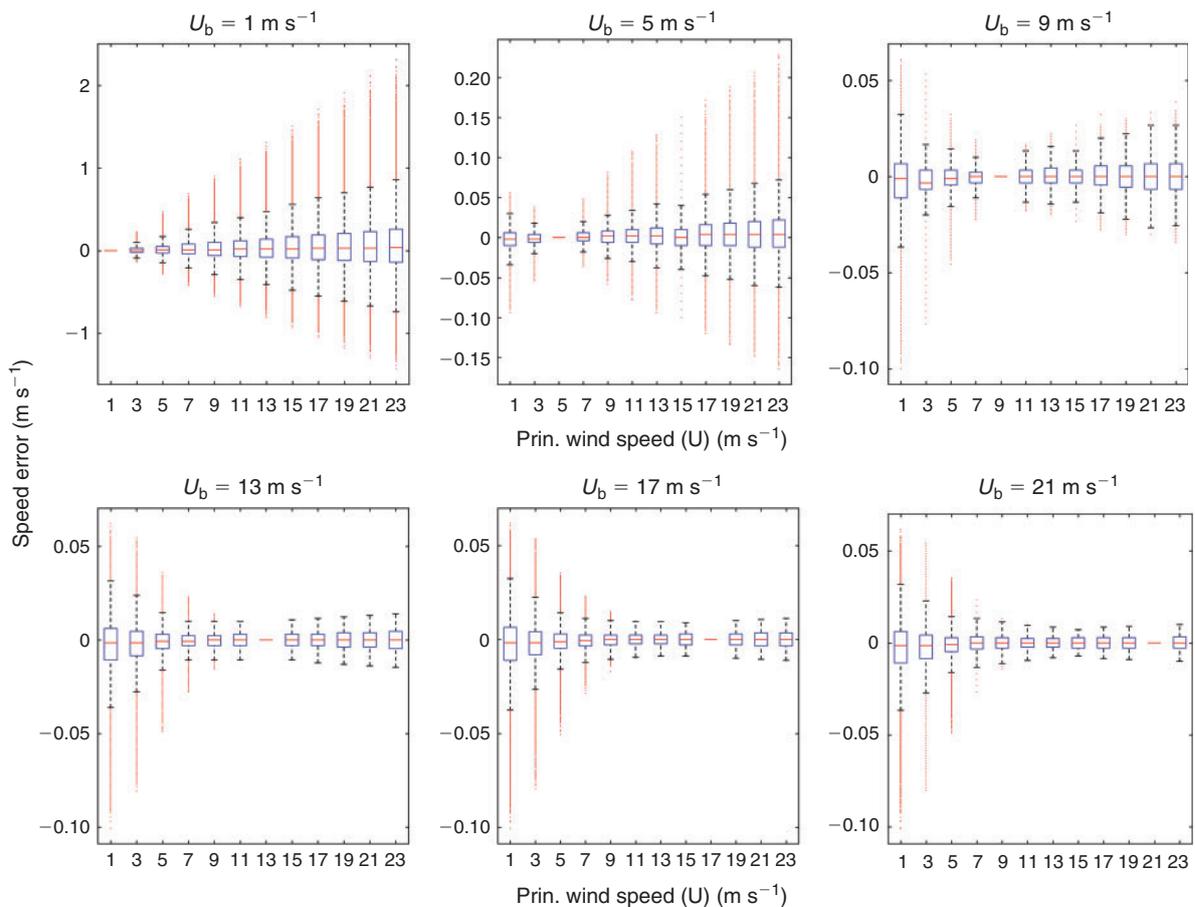
The canopy cover is grass type. See Table 1 for abbreviations. All plots are squared. The hillshade representation of each scenario is depicted in Fig. S1, available as Supplementary Material to this paper

Scenario	Res (m)	Source	Type	Altitude min–max (m)	Size (side)		$U$ ( $\text{m s}^{-1}$ ) ( $n$ )	$D$ ° ( $n$ )
					(km)	(pixels)		
Atlanta		USGS	Hill	1616–1821				
Atlanta_300×300_90m	90	–	–	–	27	300	1:2:23 (12)	0:15:359 (24)
Atlanta_900×900_30m	30	–	–	–	27	900	1:2:23 (12)	0:15:359 (24)
Atlanta_108×108_250	250	–	–	–	27	108	1:2:23 (12)	0:15:359 (24)
Boulder		USGS	Hill–Flat	1535–2615				
Boulder_108×108_250m	250	–	–	–	27	108	1:2:23 (12)	0:15:359 (24)
Boulder_300×300_90m	90	–	–	–	27	300	1:2:23 (12)	0:15:359 (24)
Boulder_900×900_30m	30	–	–	–	27	900	1:2:23 (12)	0:15:359 (24)
Idaho		USGS	Hill–Flat	123–219				
Idaho_300×300_90m	90	–	–	–	27	300	1:2:23 (12)	0:15:359 (24)
Idaho_900×900_30m	30	–	–	–	27	900	1:2:23 (12)	0:15:359 (24)
Kansas		USGS	Flat	207–286				
Kansas_108×108_250	250	–	–	–	27	108	1:2:23 (12)	0:15:359 (24)
Kansas_300×300_90m	90	–	–	–	27	300	1:2:23 (12)	0:15:359 (24)
Kansas_900×900_30m	30	–	–	–	27	900	1:2:23 (12)	0:15:359 (24)
Montseny		IGCC	Mountain	527–1972				
Montseny_50×50_90m	90	–	–	–	4,5	50	1:2:23 (12)	0:15:359 (24)
Montseny_75×75_60m	60	–	–	–	4,5	75	1:2:23 (12)	0:15:359 (24)
Montseny_150×150_30m	30	–	–	–	4,5	150	1:2:23 (12)	0:15:359 (24)
Montseny_300×300_15m	15	–	–	–	4,5	300	1:2:23 (12)	0:15:359 (24)
Montseny_900×900_5m	5	–	–	–	4,5	900	1:2:23 (12)	0:15:359 (24)
Needham		USGS	Mountain	1919–3480				
Needham_108×108_250	250	–	–	–	27	108	1:2:23 (12)	0:15:359 (24)
Needham_300×300_90m	90	–	–	–	27	300	1:2:23 (12)	0:15:359 (24)
Needham_900×900_30m	30	–	–	–	27	900	1:2:23 (12)	0:15:359 (24)
Olost		IGCC	Hill	421–641				
Olost_50×50_90m	90	–	–	–	4,5	50	1:2:23 (12)	0:15:359 (24)
Olost_75×75_60m	60	–	–	–	4,5	75	1:2:23 (12)	0:15:359 (24)
Olost_150×150_30m	30	–	–	–	4,5	150	1:2:23 (12)	0:15:359 (24)
Olost_300×300_15m	15	–	–	–	4,5	300	1:2:23 (12)	0:15:359 (24)
Olost_900×900_5m	5	–	–	–	4,5	900	1:2:23 (12)	0:15:359 (24)
Riverside		USGS	Hill	293–752				
Riverside_300×300_90m	90	–	–	–	27	300	1:2:23 (12)	0:15:359 (24)
Riverside_900×900_30m	30	–	–	–	27	900	1:2:23 (12)	0:15:359 (24)
Riverside_108×108_250	250	–	–	–	27	108	1:2:23 (12)	0:15:359 (24)

correction direction (i.e. only Eqn 3 was evaluated). The results show that speed interpolation leads to an absolute error lower than  $0.05 \text{ m s}^{-1}$  when using base maps generated with velocities higher than  $5 \text{ m s}^{-1}$ . In contrast, when using low speed maps (lower than  $5 \text{ m s}^{-1}$ ), the error can reach values similar to the actual speed (100% relative error). The explanation for this is that low speed maps introduce a high uncertainty in the interpolating factor (as highlighted in Fig. 3). Despite this fact, the overall error is low when the wind direction is similar to the base wind map. Fig. 7 shows the absolute error ( $AE^U$ ) in form of a box-plot using different base speed maps. The box represents the Q1 and Q3 quartiles (50% of the values) whereas the whisker lengths enclose 99.3% of the map's pixels. All maps generated with speeds ( $U_b$ ) higher than  $10 \text{ m s}^{-1}$  have similar errors and thus any base speed beyond this threshold is considered to be an optimum reference speed.

#### Wind direction interpolation

To assess the directional downscaling interpolation, Wind-Ninja simulations for every  $15^\circ$  were compared with interpolations generated with three different sets of base maps ( $\{W_{bi}\}$ ). Although all scenarios were processed, only the results for Boulder900×900\_30m are shown in Fig. 8. The sets were composed of: 12 maps spaced  $30^\circ$  apart (Fig. 8a), eight maps spaced  $45^\circ$  apart (Fig. 8b) and four maps at cardinal directions (Fig. 8c). The wind speed was set to  $11 \text{ m s}^{-1}$ , the same as the base maps. The mean of the absolute error is zero for all cases (error compensation) and the whiskers' length (99% of the values) increases from  $2^\circ$  when 12 base maps are used, to over  $15^\circ$  when only four directions are taken as reference. The error between base direction maps shows a similar behaviour regardless of the direction. A close look in the three panels of Fig. 8, however, shows small differences because of topography anisotropy.



**Fig. 7.** Error in wind speed for Atlanta300x300\_90m scenario when using different base speed maps and keeping the direction constant and equal to the available base map (180° at present case). Notice that y-axes for  $U_b = 1$  m s<sup>-1</sup> and  $U_b = 5$  m s<sup>-1</sup> are different from the rest for clarity. Abbreviation: Prin., principal.

*Wind speed and wind direction correction*

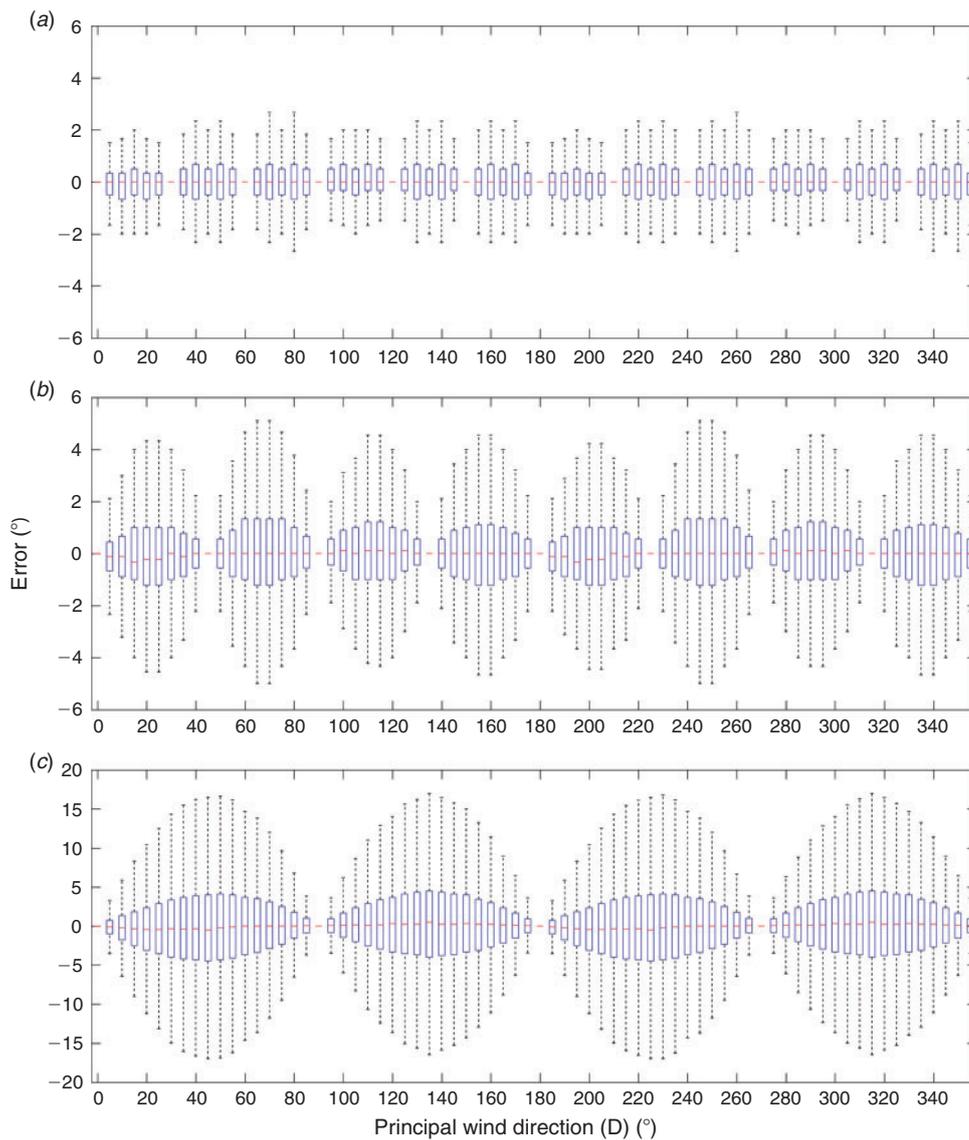
To evaluate the overall accuracy of the interpolating scheme, the principal wind direction and speed were freely changed for all 27 cases gathered in Table 2. The base maps were generated with a principal wind speed of  $U_b = 11$  m s<sup>-1</sup> and two sets of principal directions ( $\{D_{bi}\}$ ) were used: four directions (spanned 90°) and eight directions (spanned 45°). The interpolation correctness is quantified by the MAE and the standard deviation (s.d.) of every interpolated map ( $U_i, D_i$ ) compared with the WindNinja original maps ( $U_0, D_0$ ). The results are presented as filled contour plots for the wind speed ( $U$ ) and direction ( $D$ ). That is, each pixel represents a MAE (and s.d.) value for a given combination of  $U$  and  $D$ . To evaluate the interpolation strategy along multiple scenarios, they can be gathered according to DEM resolution (see Table 2) and their MAE can be averaged and shown in one figure. The results are depicted in Fig. 9 for the two scenarios of Table 2 that have 15 m resolution DEM (Fig. 9b, d) together with the eight scenarios that have 30 m resolution (Fig. 9a, c). In all cases, the direction of the compared maps span from 0 to 359°, with a step of 15° (i.e. 24 directional maps). The wind speed spans from 1 to 23 m s<sup>-1</sup>, with a step of 2 m s<sup>-1</sup> (i.e. 12 maps). The maximum error is found halfway between directional maps (either at 45° or at 22.5° from the

reference map) and increases with the wind speed up to 3 m s<sup>-1</sup> for a reference speed of 23 m s<sup>-1</sup> when the four directional maps are used, and to 1.1 m s<sup>-1</sup> at 23 m s<sup>-1</sup> on the eight maps scheme.

The wind direction (Fig. 10) is less correctly solved. The mean absolute error grows beyond 10° with a standard deviation that reaches 20° in the worst case (reference maps spacing 90°). When more base maps are used, or the DEM resolution is decreased, the MAE together with the maximum standard deviation is decreased to 6°. Doubling the amount of available base maps halves the incurred error. In all cases, the error is constant along the reference wind direction. This indicates that the directional interpolation uncertainty dominates over the wind speed interpolation uncertainty. Figs S1, S2 and S3, available as Supplementary Material to this paper, show the APE (absolute percentage error) to further support this discussion.

*Fire spread comparison*

To assess the quality of the interpolating scheme, we used the fire perimeter spread simulator described in (Rios et al. 2016) with added topography to perform a 30 min synthetic run over the Montseny scenario (Fig. 11). This data-driven simulator uses a Rothermel approach as a core fire spread model. The perimeter evolution is compared when using interpolated downscaling maps,



**Fig. 8.** Box-plot comparison of directional downscaling error when changing the step to generate the set of base maps. Notice that the  $y$ -axis of the bottom panel differs from the two other for clarity.

original WindNinja maps and constant wind. This comparison is not intended to validate the propagation model, but to assess the application of the proposed interpolation framework and to highlight the improvement when using near-surface wind fields.

To represent the worst case, base maps were generated at  $U_b = 5 \text{ m s}^{-1}$  with two different sets of directions ( $\{D_{bi}\}$ ): four and eight directions (i.e. every  $90^\circ$  and  $45^\circ$ ). The updated principal wind was set to have a speed of  $U = 10 \text{ m s}^{-1}$  blowing from  $240^\circ$  (towards north-east). These conditions correspond to the highest error committed by the interpolation, as discussed in previous sections. A  $15 \text{ m}$  resolution and a DEM with  $600 \times 600$  cells (i.e.  $9 \times 9 \text{ km}$ ) was used. The same resolution was used for the wind maps. The corresponding results are shown in Fig. 11. Isochrones ( $120 \text{ s}$  apart) are projected over the DEM hillshade map. Both interpolated and original maps allow the simulator to capture near-surface effects better than the homogeneous wind case, as

the wind is influenced by topography. In the homogeneous wind case, the overall terrain influence is not represented. This is clearly visible on the two sides of the ridge (at the centre of the image in Fig. 11). The fire front has difficulties to propagate downhill on the lee side and it breaks into two subfronts, separated by a lower rate-of-spread zone. This is particularly affected by a region with almost no wind (see Fig. 12).

Differences of up to  $250 \text{ m}$  in the fire front location exist at  $30 \text{ min}$  between the WindNinja map and the interpolated map when four reference maps are used (Fig. 11a). This is because of the fact that with the interpolated maps the spread model does not correctly simulate the wind field on the lee side of the ridge, and as a consequence a stagnation point develops at the front. This lee side effect is not captured by the simulation with a homogeneous wind field and remains mostly unresolved by a  $90^\circ$  step interpolation scheme. On the flanks the difference

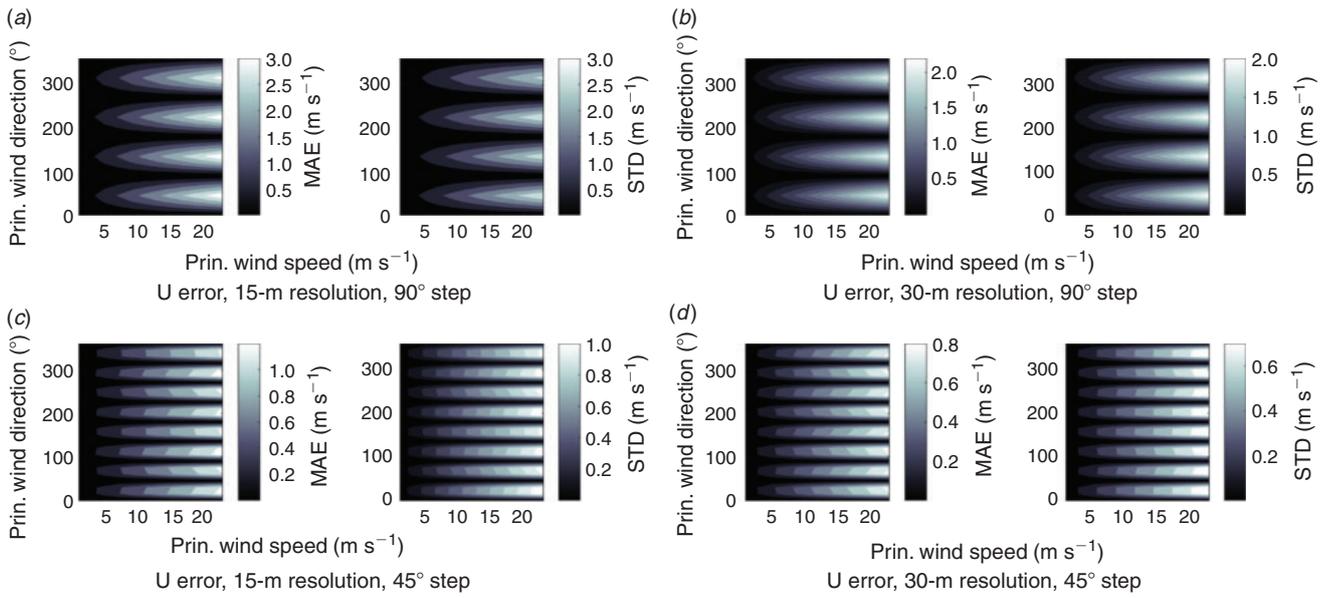


Fig. 9. Wind speed validation for all 15 and 30 m resolution scenarios. The mean absolute error (MAE) and standard deviation (s.d.) are represented.

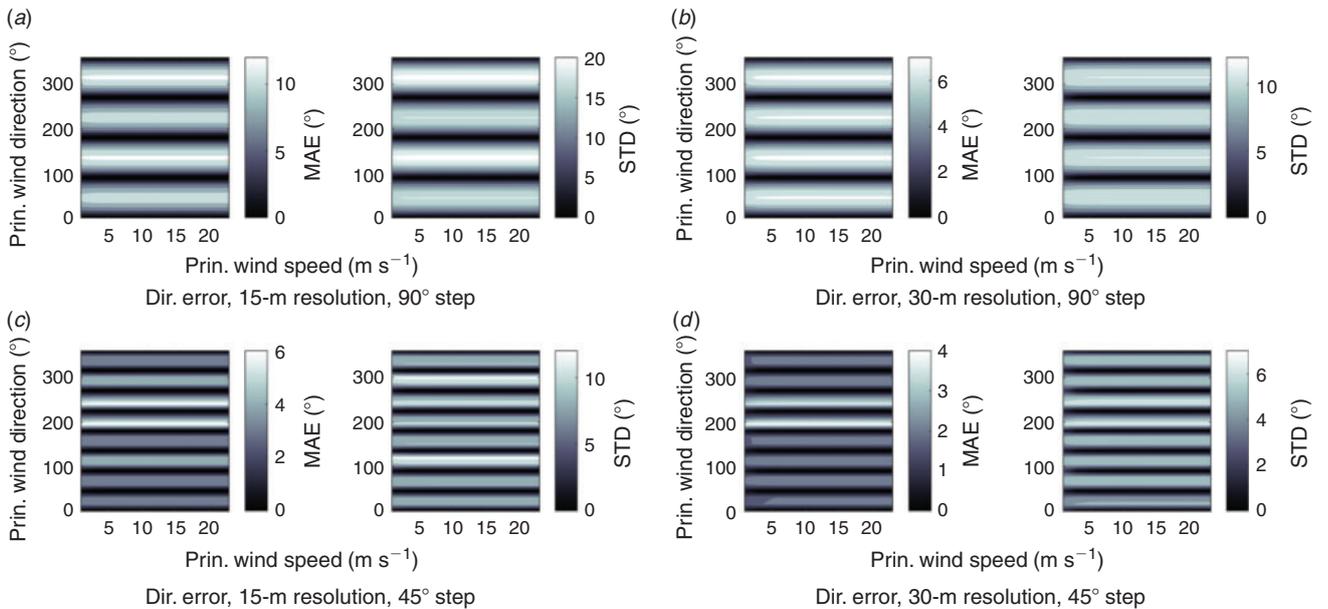


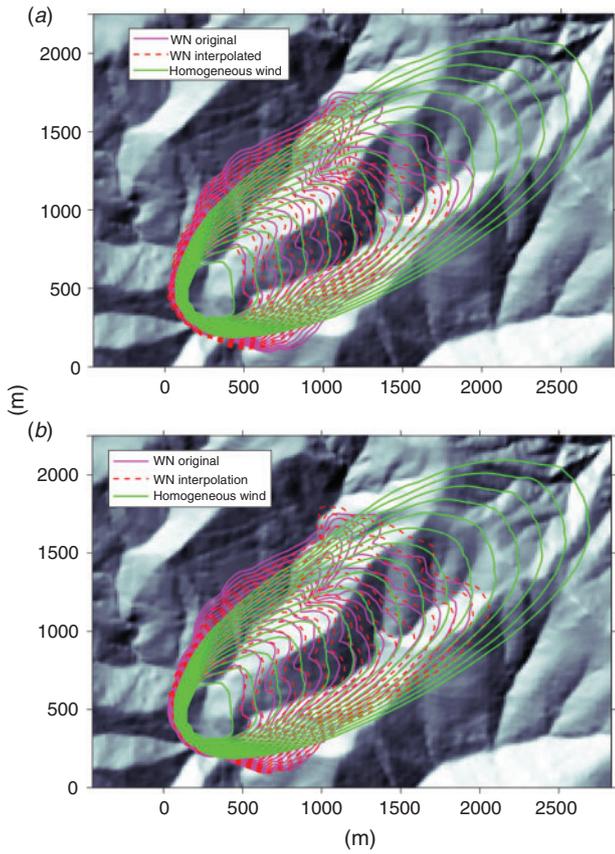
Fig. 10. Wind direction validation for all 15 and 30 m resolution scenarios. The mean absolute error (MAE) and s.d. are represented.

between original maps and interpolated maps is reduced to less than 20 m.

When eight reference wind field maps are used (Fig. 11b), the lee side is properly resolved, so that no stagnation point appears, and the differences in the fire front are between 10 and 80 m. The right flank correctly matches the original runs, whereas on the left flank the error grows but remains below 10 m for the last isochrone (i.e. after 30 min of propagation). To better illustrate the difference between fire spread runs (differences lower than 10 m), individual isochrone plots (i.e. a pair of isochrones at a time) are shown in Figs S5–S7).

In order to quantitatively assess the performance of the wind interpolating scheme, a front-to-front comparison was performed. For this, the shape deviation index (SDI) was calculated for every isochrone and displayed in Fig. 13. The  $SDI_t$  is an error metric commonly used to compare front agreement at a given time  $t$ , and it is defined as (Cui and Perera 2010; Rios et al. 2016):

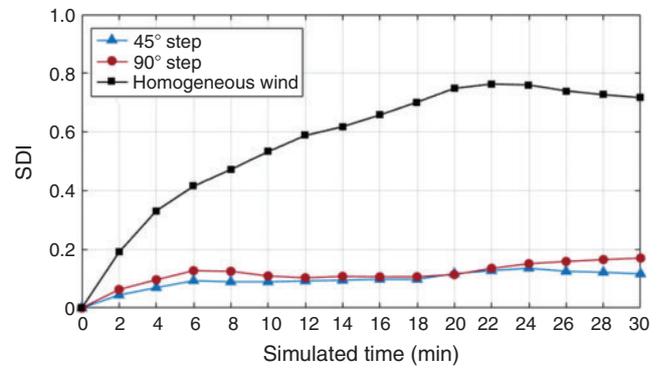
$$SDI_t = 1 - \frac{A_t^o \cap A_t^i}{A_t^o} \quad (9)$$



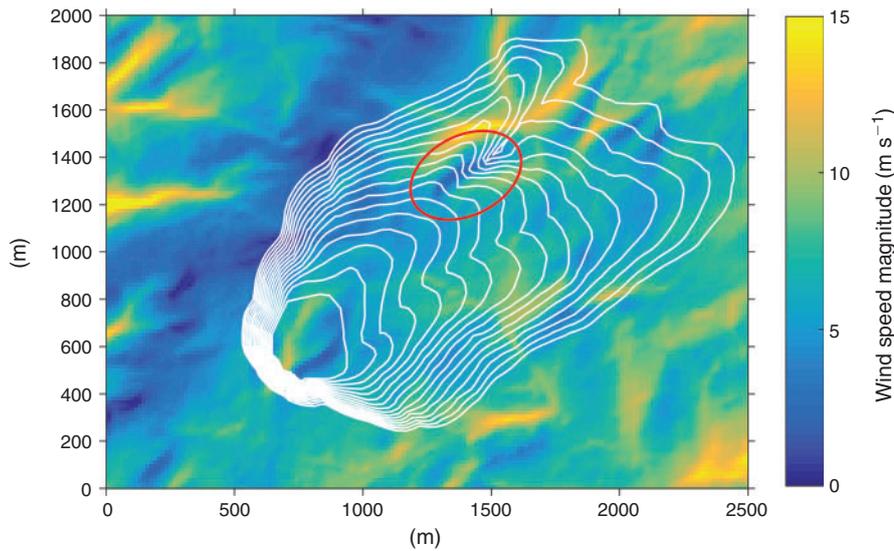
**Fig. 11.** Comparison of a 30 min synthetic run with input wind speed of  $5 \text{ m s}^{-1}$  and wind blowing from  $240^\circ$  with respect to north. Spread using WindNinja (WN) maps (magenta solid lines) is depicted every 2 min. Red dashed lines are the fronts generated applying the interpolating framework with different base maps settings. Green solid lines are the propagation with homogeneous wind field. The hillshade map is displayed in the background.

where  $A_t^o$  is the area of the isochrone using an original WindNinja wind field at time ( $t$ ) and  $A_t^i$  is the area of the corresponding isochrone produced with the interpolating scheme. Note that as a better match is achieved, the SDI value tends towards zero. Fig. 13 clearly reflects the improvement of using the interpolation framework compared with using a homogeneous wind field, as the error is kept below 20% for a step size of  $90^\circ$  and 15% for a step size of  $45^\circ$ .

Differences between both interpolation strategies begin to grow after 20 min, and they are on the order of 20 m on the south-eastern part of the fire front. Each wind field map simulated with WindNinja takes  $\sim 50 \text{ s}$  to run when parallelised on an eight processor (Intel Xeon 2.3 Ghz E5-2697 ver. 4) dedicated workstation. Applying the interpolating framework presented in this paper gives an almost instant solution (run time  $< 10^{-3} \text{ s}$ , without taking into account the run time of the pre-run base maps). The computational break-even point is then at four



**Fig. 13.** Similarity index comparison for fire fronts simulated with a homogeneous wind field (black squared markers), interpolation using  $90^\circ$  step (round red markers) and  $45^\circ$  step (blue triangles).



**Fig. 12.** Fire spreading simulated perimeters run with a WindNinja (WN) original wind map displayed over the wind magnitude ( $\|\vec{U}\|$ ) map. The lee side stagnation region is highlighted with a red circle.

(eight, if eight base maps are wanted) WindNinja iterations. That is, for any application that requires more than four (eight) updates of the wind field, the interpolation framework at hand is faster at a ratio of a minute per additional wind map update required. Given that the typical number of required wind map updates in data-driven applications is of the order of several hundred, this is clearly a major improvement.

### Conclusions

A framework for downscaling and interpolating high-resolution near-surface wind fields is presented and validated in this article. The dependence of the downscaling factors is explored statistically in terms of wind speed, wind direction, vegetation structure and DEM resolution. The overall system is evaluated with WindNinja simulations performed on eight different scenarios representing diverse topographic configurations. Results showed that the wind speed used to generate reference wind maps does not have a considerable influence as long as it is high enough to capture the main near-surface wind flows (i.e.  $>3 \text{ m s}^{-1}$ ). The base wind speed can thus be simply extrapolated. Wind direction, however, has a larger influence on downscaled wind maps and must therefore be interpolated. Interpolations were compared using two reference wind direction sets (with four and eight base direction maps). The wind speed MAPE grows up to 7% for  $20 \text{ m s}^{-1}$  when using only four base maps and the directional error has a maximum MAE of  $5^\circ$ . When using base maps for wind directions at every  $45^\circ$ , the wind speed MAPE decreases to 3% and the directional MAE decreases to  $3^\circ$ . A 30 min fire spread simulation was used to compare and validate the two interpolating framework configurations. The present implementation offers a significant reduction in computing time if multiple wind simulations are to be run in a particular scenario (as it would be the case in data assimilation procedures). This speed-up can be in the order of hours for optimisation routines requiring a high number of model evaluations (e.g. genetic algorithms and gradient-based methods). If eight base directions are used, the discrepancy with respect to WindNinja simulations is kept low enough so that the fire spread simulations initialised with those interpolated data are equally acceptable for operational purposes (less than 80 m discrepancy in the fire front, and less than 10 m on the flanks after 30 min propagation). The results prove this methodology is reliable and capable of speeding up recursive near-surface wind evaluations, which would enable data assimilation and probabilistic risk assessment applications currently unfeasible.

### Conflicts of interest

The authors declare no conflicts of interest.

### Acknowledgements

The authors thank the Spanish Ministry of Economy and Competitiveness (project CTM2014-57448-R), the Spanish Ministry of Education, Culture and Sport (Formación de Profesorado Universitario Program) and the Autonomous Government of Catalonia (project no. 2014-SGR-413). The second author acknowledges the funding by the Chilean National Research Center for Integrated Natural Disaster Management (CIGIDEN) through Grant CONICYT/FONDAP/15110017.

### References

- Albini FA (1982) Response of free-burning fires to nonsteady wind. *Combustion Science and Technology* **29**, 225–241. doi:10.1080/00102208208923599
- Altintas I, Block J, De Callafon R, Crawl D, Cowart C, Gupta A, Nguyen M, Braun H-W, Schulze J, Gollner M, Trouve A, Smarr L (2015) Towards an integrated cyberinfrastructure for scalable data-driven monitoring, dynamic prediction and resilience of wildfires. *Procedia Computer Science* **51**, 1633–1642. doi:10.1016/j.PROCS.2015.05.296
- Butler BW, Wagenbrenner NS, Forthofer JM, Lamb BK, Shannon KS, Finn D, Eckman RM, Clawson K, Bradshaw L, Sopko P, Beard S, Jimenez D, Wold C, Vosburgh M (2015) High-resolution observations of the near-surface wind field over an isolated mountain and in a steep river canyon. *Atmospheric Chemistry and Physics* **15**, 3785–3801. doi:10.5194/ACP-15-3785-2015
- Ching J, Rotunno R, LeMone M, Martilli A, Kosovic B, Jimenez PA, Dudhia J (2014) Convectively Induced secondary circulations in Fine-Grid Mesoscale Numerical Weather Prediction Models. *Monthly Weather Review* **142**, 3284–3302. doi:10.1175/MWR-D-13-00318.1
- Clark TL, Coen J, Latham D (2004) Description of a coupled atmosphere-fire model. *International Journal of Wildland Fire* **13**, 49–63. doi:10.1071/WF03043
- Cui W, Perera AH (2010) Quantifying Spatio-Temporal errors in forest fire spread modelling explicitly. *Journal of Environmental Informatics* **16**, 19–26. doi:10.3808/JEI.201000174
- Filippi JB, Bosseur F, Pialat X, Santoni PA, Strada S, Mari C (2011) Simulation of coupled fire/atmosphere interaction with the MesoNH-ForeFire models. *Journal of Combustion* **2011**, 540390. doi:10.1155/2011/540390
- Finney M (1998) FARSITE: fire area simulator: model development and evaluation. (USDA Forest Service, Rocky Mountain Research Station: Ogden, UT, USA) Available at [https://www.fs.fed.us/rm/pubs/rmrs\\_rp004.pdf](https://www.fs.fed.us/rm/pubs/rmrs_rp004.pdf) [Verified 28 March 2018]
- Finney M (2006) An overview of FlamMap fire modeling capabilities. In 'Fuel management – how to measure success: Conference proceedings', 28–30 March 2006. USDA Forest Service, Rocky Mountain Research Station, Research Paper RMRS-P-41, pp. 213–220. (Ogden, UT)
- Forthofer JM (2007) Modeling wind in complex terrain for use in fire spread prediction. Masters thesis, Colorado State University, Fort Collins, CO, USA.
- Forthofer J, Shannon K, Butler B (2009) Simulating diurnally driven slope winds with WindNinja. In 'Proceedings of the 8th symposium on fire and forest meteorology', 13–15 October 2009, Kalispell, MT, USA. American Meteorological Society. Available at <https://ams.confex.com/ams/pdfpapers/156275.pdf> [Verified 24 March 2018]
- Forthofer JM, Butler BW, Wagenbrenner NS (2014a) A comparison of three approaches for simulating fine-scale surface winds in support of wildland fire management. Part I. Model formulation and comparison against measurements. *International Journal of Wildland Fire* **23**, 969–981. doi:10.1071/WF12089
- Forthofer JM, Butler BW, Mchugh CW, Finney M, Bradshaw LS, Stratton RD, Shannon KS, Wagenbrenner NS (2014b) A comparison of three approaches for simulating fine-scale surface winds in support of wildland fire management. Part II. An exploratory study of the effect of simulated winds on fire growth simulations. *International Journal of Wildland Fire* **23**, 982–994. doi:10.1071/WF12090
- Homicz G (2002) Three-dimensional wind field modeling: a review. Sandia National Laboratories, Albuquerque, SANDIA (August). Available at <http://prod.sandia.gov/techlib/access-control.cgi/2002/022597.pdf> [Verified 24 March 2018]
- Lopes A (2003) WindStation – a software for the simulation of atmospheric flows over complex topography. *Environmental Modelling & Software* **18**, 81–96. doi:10.1016/S1364-8152(02)00024-5

- Lopes AMG, Cruz MG, Viegas DX (2002) Firestation – an integrated software system for the numerical simulation of fire spread on complex topography. *Environmental Modelling & Software* **17**, 269–285. doi:10.1016/S1364-8152(01)00072-X
- Lundquist KA, Chow FK, Lundquist JK (2010) An immersed boundary method for the weather research and forecasting model. *Monthly Weather Review* **138**, 796–817. doi:10.1175/2009MWR2990.1
- Mandel J, Beezley JD, Coen JL, Kim M (2009) Data assimilation for wildland fires. *Control Systems, IEEE* **29**, 47–65. doi:10.1109/MCS.2009.932224
- Monedero S, Buckley D, Ramírez J (2011) New approaches in fire simulations analysis with wildfire analyst. Available at <http://dx.doi.org/10.13140/2.1.2045.7766> [Verified 31 March 2018]
- Rios OWJ, Rein G (2014) Forecasting wind-driven wildfires using an inverse modelling approach. *Natural Hazards and Earth System Sciences* **14**, 1491–1503. doi:10.5194/NHESS-14-1491-2014
- Rios O, Pastor E, Valero MM, Planas E (2016) Short-term fire front spread prediction using inverse modelling and airborne infrared images. *International Journal of Wildland Fire* **20**, 1015–1032.
- Rochoux M, Emery C, Ricci S, Cuenot B, Trouvé A (2014) Towards predictive simulation of wildfire spread at regional scale using ensemble-based data assimilation to correct the fire front position. *Fire Safety Science* **11**, 1442–1456. doi:10.3801/IAFSS.FSS.11-1443
- Rothermel RC (1972) A mathematical model for predicting fire spread in wildland fuels. USDA Forest Service, Intermountain Forest and Range Experiment Station, Research Paper INT-RP-115. (Ogden, UT)
- Sanjuan G, Margalef T, Cortés A (2016a) Applying domain decomposition to wind field calculation. *Parallel Computing* **57**, 484–490. doi:10.1109/HPCSIM.2015.7237080
- Sanjuan G, Margalef T, Cortés A (2016b) Hybrid application to accelerate wind field calculation. *Journal of Computational Science* **17**, 576–590. doi:10.1016/J.JOCS.2016.07.011
- Sanjuan G, Tena C, Margalef T, Cortés A (2016c) Applying vectorization of diagonal sparse matrix to accelerate wind field calculation. *Journal of Supercomputing* **73**, 240–258. doi:10.1007/S11227-016-1696-9
- Seaman NL, Gaudet BJ, Stauffer DR, Mahrt L, Richardson SJ, Zielonka JR, Wyngaard JC (2012) Numerical prediction of submesoscale flow in the nocturnal stable boundary layer over complex terrain. *Monthly Weather Review* **140**, 956–977. doi:10.1175/MWR-D-11-00061.1
- Sharples JJ, McRae RHD, Wilkes SR (2012) Wind–terrain effects on the propagation of wildfires in rugged terrain: fire channelling. *International Journal of Wildland Fire* **21**, 282–296. doi:10.1071/WF10055
- Viegas DX, Simeoni A (2011) Eruptive behaviour of forest fires. *Fire Technology* **47**, 303–320. doi:10.1007/S10694-010-0193-6
- Wagenbrenner NS, Forthofer JM, Lamb BK, Shannon KS, Butler BW (2016) Downscaling surface wind predictions from numerical weather prediction models in complex terrain with WindNinja. *Atmospheric Chemistry and Physics* **16**, 5229–5241. doi:10.5194/ACP-16-5229-2016
- Weller HG, Tabor G (1998) A tensorial approach to computational continuum mechanics using object-oriented techniques. *Computers in Physics* **12**, 620–631. doi:10.1063/1.168744
- Wyngaard JC (2004) Toward numerical modeling in the ‘Terra Incognita’. *Journal of the Atmospheric Sciences* **61**, 1816–1826. doi:10.1175/1520-0469(2004)061<1816:TNMITT>2.0.CO;2
- Zhang C, Rochoux M, Tang W, Gollner M, Filippi JB, Trouvé A (2017) Evaluation of a data-driven wildland fire spread forecast model with spatially distributed parameter estimation in simulations of the FireFlux I field-scale experiment. *Fire Safety Journal* **91**, 758–767. doi:10.1016/J.FIRESAF.2017.03.057