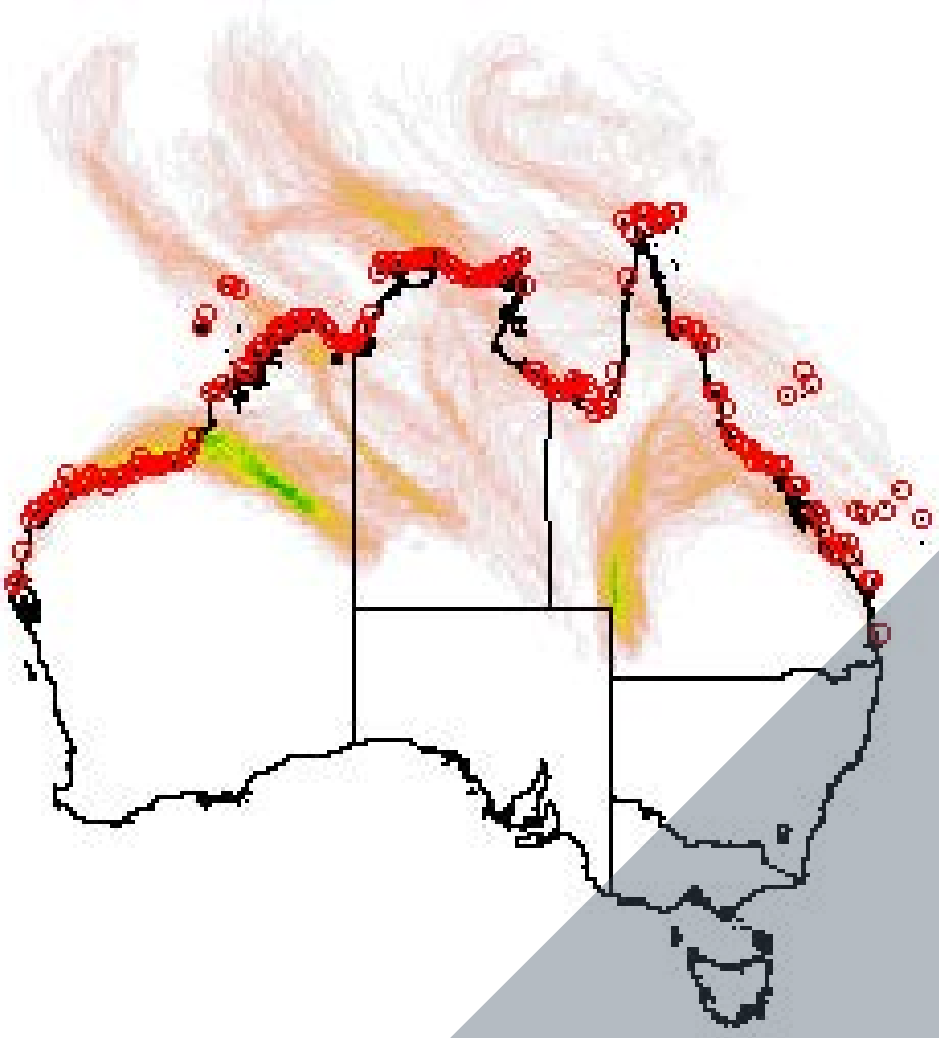


Rapid real-time simulation of wind-assisted long-ranged dispersal of FAW in Australia

Final Technical Report

July 2023

cesar

Plant Health AUSTRALIA

Australian Government

Department of Agriculture, Fisheries and Forestry

Location: Level 1
1 Phipps Close
DEAKIN ACT 2600
Phone: +61 2 6215 7700
Email: biosecurity@phau.com.au
Visit our website planthealthaustralia.com.au

An electronic copy of this plan is available through the email address listed above.

In referencing this document, the preferred citation is:

Plant Health Australia Ltd (2023) Rapid real-time simulation of wind-assisted long-ranged dispersal of fall armyworm in Australia. Plant Health Australia, Canberra, ACT.

This project has been funded by the Department of Agriculture, Fisheries and Forestry through its *"Boosting national interest research and development for Australia's ongoing management of Spodoptera frugiperda (Fall armyworm)"* program with Plant Health Australia. For more information about this program visit planthealthaustralia.com.au/fall-armyworm/

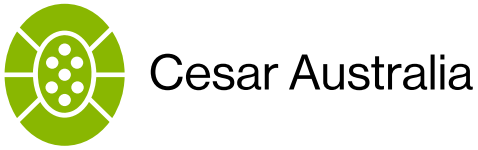
Disclaimer:

The material contained in this publication is produced for general information only. It is not intended as professional advice on any particular matter. No person should act or fail to act on the basis of any material contained in this publication without first obtaining specific and independent professional advice.

Plant Health Australia and all persons acting for Plant Health Australia in preparing this publication, expressly disclaim all and any liability to any persons in respect of anything done by any such person in reliance, whether in whole or in part, on this publication. The views expressed in this publication are not necessarily those of Plant Health Australia.

Acknowledgements

This project *Rapid real-time simulation of wind-assisted long-ranged dispersal of fall armyworm in Australia* was undertaken by Cesar Australia, managed by Plant Health Australia and funded by the Department of Agriculture, Fisheries and Forestry through the program *Boosting national research and development for Australia's ongoing management of Fall Armyworm.*



Cesar Australia

Rapid real-time simulation of wind-assisted long-ranged dispersal of fall armyworm in Australia

Alex Slavenko & Luis Mata

30 June 2023

Prepared for

Plant Health Australia Ltd

Version history

Version	Date	Description	Author(s)	Reviewed by
1.0	24/05/2023	Final Report_Draft	A Slavenko	L Mata
1.1	30/06/2023	Final Report	A Slavenko; L Mata	A Slavenko

Project team

Team member	Role
Dr. Alex Slavenko	Project Consultant
Dr. Luis Mata	Project Supervisor

Contents

Executive summary 4

Background 5

Methods 6

- Cellular automata model for wind-assisted long-distance dispersal 6
- Comparison with the NOAA HYSPLIT simulations 8
- Validation against empirical data on FAW migration 10

Results 12

- Comparison with the NOAA HYSPLIT simulations 12
- Validation against empirical data on FAW migration 19
- Web tool 24

Conclusion 26

Acknowledgments 26

References 28

Appendix 1. R code to run CA wind-assisted dispersal model 31

Abbreviations

Abbreviation	Description
ANOVA	Analysis of variance
AUC	Area under the curve
CA	Cellular automata
FAW	Fall armyworm, *Spodoptera frugiperda*
GAM	Generalised additive model
GFS	NOAA global forecast system
GIS	Geographic information system
HYSPLIT	Hybrid Single-Particle Lagrangian Integrated Trajectory model
NOAA	National Oceanic and Atmospheric Administration (United States)
QDAF	Queensland Department of Agriculture and Fisheries
ROC	Receiver operating characteristic

Executive summary

The fall armyworm (FAW, *Spodoptera frugiperda*) is a lepidopteran pest recently established in Australia. It feeds in large numbers on the leaves, stems and reproductive parts of more than 350 plant species, causing major damage to economically important crops. As this pest can migrate large distances over short periods of time, regular monitoring and surveillance are vital. Knowing when the adult moths are likely to appear in an area helps to inform monitoring for larvae and proactive management if required.

We have created a novel wind-assisted long-distance dispersal model, incorporating a spatially explicit cellular automata algorithm. The model can be used to simulate dispersal trajectories of FAW based on weather forecast data. We have implemented the model in a friendly web application, enabling users to make forwards predictions of FAW movement from known populations as an early warning system, or to make backwards predictions to identify the potential source of a newly discovered population.

This dispersal model will add to the expanding toolkit of growers, agronomists, and researchers in devising optimal management strategies to counter the threat posed by this economically important pest species.

Background

The fall armyworm (FAW, *Spodoptera frugiperda*) is a lepidopteran pest that feeds in large numbers on the leaves, stems and reproductive parts of more than 350 plant species, causing major damage to economically important grains such as maize, rice, sorghum, sugarcane and wheat, as well as other vegetable crops and cotton. Native to the Americas, it was first reported in Africa in January 2016, where it is now widely established. It was subsequently reported in the Middle East and Asia in 2018. In Australia, it was reported in January 2020 in the Torres Strait and subsequently discovered in Queensland in February 2020. By February 2021, FAW was detected in NSW and northern Victoria.

The large dispersal potential of FAW adults, wide host range of immature feeding stages, and unique environmental conditions in its invasive range creates large uncertainties in the expected impact on Australian plant production industries and natural environments. Enhanced predictability of FAW population movement across large spatial scales will support targeted surveillance, monitoring and management, and thus reduce the impact of FAW in Australia. While some preliminary preparedness work on FAW has been undertaken in Australia, including a predictive model for seasonal activity potential (Maino et al. 2021), this model did not consider the role of wind-assisted long-distance dispersal, and made the simplifying assumption of random long-distance dispersal.

The role of atmospheric conditions on the long-range dispersal and deposition of FAW has been shown to increase predictability of migrating populations (Westbrook and Sparks 1986; Mitchell et al. 1991; Westbrook 2008; Westbrook et al. 2016; 2019). A diverse range of atmospheric processes are known to facilitate insect transport (Drake and Farrow 1988). In Australia, with a few exceptions (e.g., the south-east trade winds affecting coastal northern Queensland or the winter westerlies of southern Australia), there is a lack of seasonal prevailing winds (Gregg et al. 2001). Instead, Australia typically sees sporadic and short-lived winds that are favourable for migration (i.e., sufficiently strong, and warm). Especially in spring, but also in summer, these are often northerlies or north-westerlies, ahead of cold fronts. In summer, there are also the post-frontal south-westerlies. Thus, in Australia, long-range wind-borne movement is a more stochastic process, depending on migration-ready populations, local weather conducive to nocturnal take-off and climbing to high altitudes, and atmospheric transport mechanisms capable of carrying FAW across long distances. In the southern half of Australia in spring, these conditions are often associated with the passage of cold fronts, and the window of opportunity may be as short as a week (Drake 1994). The presence or absence of these conditions can determine whether, in any given season, moths will be found outside the range predicted by dispersal models such as ours, or conversely, not found in locations where the model predicts they should be. It will be important to determine whether FAW can exploit the same opportunities for long range movement used by some endemic lepidopteran pest species, such as the cotton bollworm (*Helicoverpa punctigera*; Drake 1994; Gregg et al. 2001).

Methods

Recently, preliminary predictive modelling research on the seasonal occurrence of fall armyworm in Australia was undertaken through the GRDC project (CES2004-003RTX) "Preparedness and Management for fall armyworm (*Spodoptera frugiperda*)". Notably, this preliminary model made the simplifying assumption that long-distance dispersal was random. In the current project, we relaxed this simplifying assumption of random long-distance dispersal through the development of a novel wind-assisted long-distance dispersal model. Specifically, we (1) compared our novel model to the existing HYSPLIT model; (2) validated model predictions against observations of FAW movements throughout eastern Australia in 2020-2021; and (3) developed a user-friendly web tool for easy application of the model, providing industry users with streamlined access to the model's features and functionalities.

Cellular automata model for wind-assisted long-distance dispersal

The HYSPLIT model computes air parcel trajectories and deposition, or dispersion of atmospheric pollutants (Stein et al. 2015; Rolph et al. 2017). It was jointly developed by the United States' National Oceanic and Atmospheric Administration (NOAA) and Australia's Bureau of Meteorology (Draxler and Hess 1998). Nightly emigration flights of FAW have been previously simulated by the HYSPLIT physical model (Westbrook et al. 2019). While HYSPLIT represents the state-of-the-art in wind dispersion models, it currently has several limitations. The model is complex and computationally intensive, does not consider biological processes such as reproduction, nor other climatic factors that may be limiting species survival and development (e.g., temperature). The complexity of physically based models, lack of biology, and inability to produce fast grid-based predictions make them unsuitable as an efficient early warning system.

Here, we generated a simple but efficient cellular automata (CA) algorithm to simulate the potential migration paths and daily location of FAW. CA is a spatially explicit technique that incorporates spatial interactions between cells and their neighbourhood to generate a global pattern. The CA technique offers a versatile method for modelling complex physical systems using simple operations (Wolfram 1984). This simplification dramatically reduces the computational load of a CA model compared with physically based models such as HYSPLIT. Another key advantage of the CA model over HYSPLIT is the capacity of the former to forecast/hindcast over a grid with multiple locations or regions as the source populations. This feature may be used to extend preliminary Australian FAW models that already consider the biology of the species, seasonal population dynamics, and regional climatic suitability.

We used data from the NOAA global forecast system (GFS) to develop the data pipeline and for the base development of the spatial and spatio-temporal CA method. The GFS data includes daily wind U and V components (representing longitudinal and latitudinal air movement, respectively) at different atmospheric levels from the Earth's surface to higher levels. The dataset has a 6-hour temporal resolution (i.e., it is updated daily and four times per day at 0, 6, 12, and 18 hours UTC) at a 0.25 x 0.25 degrees spatial resolution (roughly 25 x 25 km).

We wrote a pipeline in the R programming language v4.2.1 (R Core Team 2022) to download the daily GFS data. For each forecast cycle we download forecasts for the subsequent 48 hours, which can be then used to initiate the simulations. The U and V wind components are downloaded at 850 and 950 mb atmospheric levels (approximately 500 and 1400 m above sea level), as these are linked to long-distance insect migration (Greenslade et al. 1999; Westbrook et al. 2019). They are then used to calculate wind speed and direction using mathematical operations (Figure 1).

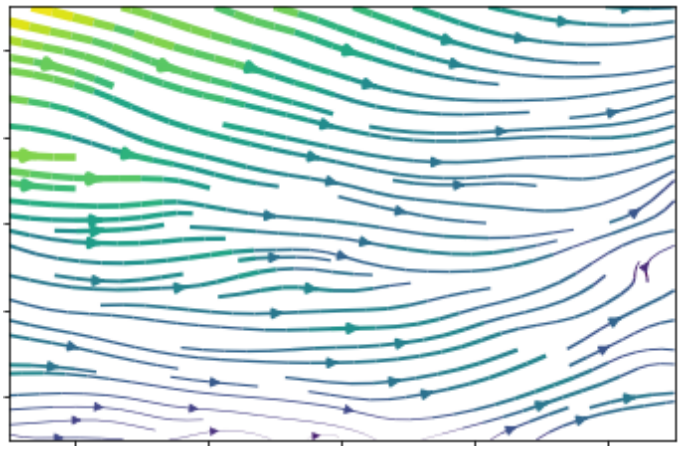


Figure 1: Estimated wind direction (arrows) and wind speed (arrow thickness) based on U and V wind components. Source: GFS wind data.

The simulation model then implements a CA algorithm to simulate the movement of particles with wind over a specific period (hourly steps of up to 48 hours) from a given starting location. The algorithm uses the calculated speed and direction in each grid cell at each step of simulation to simulate the movement of a particle through the grid.

In each step, the dispersal distance is calculated based on wind speed and cell size, and dispersal direction is calculated based on wind direction plus a random change in the direction of wind, which is drawn from a uniform distribution (Uniform [-30, 30]; units: decimal degrees). A backwards hindcast functionality is also available, wherein the computed direction is reversed, and hourly steps are counted backwards.

Multiple simulations can be run with the same starting conditions, and from multiple starting locations simultaneously, resulting in a map of frequencies of simulated trajectories per grid cell, representing the probability of wind-assisted dispersal to each cell (Figure 2). Simulation outputs can be exported as GIS grid formats that can easily integrated into mechanistic biosecurity and ecological models (Maino et al. 2021). R code to run the simulation model is available in Appendix 1.

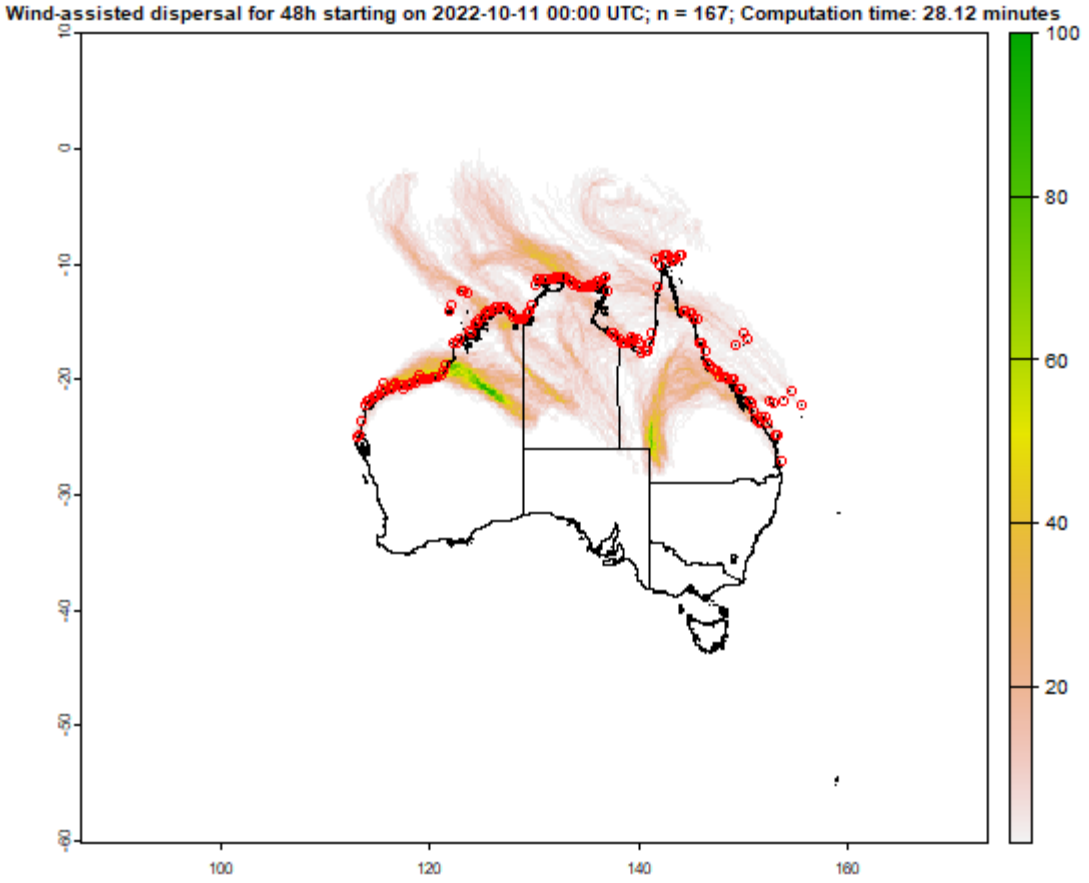


Figure 2: Map showing the frequency of particle trajectories from 167 starting points (marked in red) crossing 0.25-degree cells in Australia. Frequency is colour-coded from 0 (white) to 100% (green).

Comparison with the NOAA HYSPLIT simulations

To check the accuracy of the CA model we compared our results with the HYSPLIT particle diffusion model. We ran both models for 100 randomly selected locations across Australia on the same meteorological forecast cycles derived from the NOAA GFS. The following parameters were used:

1. Start date: 11/10/2022 00:00 UTC.
2. Forecast period: 48 hours.
3. 0.25 degrees spatial resolution (roughly 25 x 25 km).
4. Height: starting at 500 m (HYSPLIT), 950 mb (CA)

For each starting point we simulated the particle trajectory using the HYSPLIT model, followed by 50 simulated trajectories using the CA model to account for the stochastic element in the CA model calculations. We then calculated the following two metrics for each trajectory in each timestep:

1. Distance travelled, calculated as the greater circle distance between starting point and location at current timestep, in km.
2. Azimuth between starting point and location at current timestep, with north = 0^{o}.

Additionally, for the HYSPLIT models we recorded the elevation at each timestep in atmospheric pressure [mb] (since the HYSPLIT model, unlike the CA model, allows vertical movement of particles).

We used two different and complimentary approaches to compare the output of the CA model to the HYSPLIT model:

1. We generated distributions of trajectory metrics from the 50 CA simulations and performed one-sample t-tests to compare to a population mean μ equal to the metric from the corresponding HYSPLIT trajectory. Thus, we tested the null hypothesis that the CA trajectories are sampled from a distribution centred around the HYSPLIT trajectory. We then tallied, for each metric, the number of starting points in which the null hypothesis was rejected (i.e., the trajectories of the CA simulations differ from the HYSPLIT trajectory) and the number of starting points in which the null hypothesis was not rejected (i.e., the trajectories of the CA simulations do not differ from the HYSPLIT trajectory). We ran exact binomial tests to see if the number of "successes" (null not rejected) out of the 100 starting points was higher than expected by chance.
2. We subtracted each metric of the HYSPLIT trajectory from all 50 simulated CA trajectory metrics. To estimate differences in azimuth, we calculated the offset between CA and HYSPLIT trajectories by taking the absolute value of subtracting the HYSPLIT azimuth from the CA azimuth. This is equivalent to the clockwise angle of the HYSPLIT trajectory compared to the CA trajectory. We converted this offset value to radians. We then fit a generalised additive model (GAM) using the R function 'bam' (suitable for large datasets) from the *mgcv* package v1.8.40 (Wood 2011), with both forecast hour and spatial coordinates of the starting point modelled using thin plate splines (Wood 2003). The GAM is specified as:

   $$y \sim s(t) + s(x_n, y_n) + s(h)$$

   where y is the difference in the examined metric, t is the hour of forecast, n is the starting point, and h is the elevation (in mb) of the HYSPLIT trajectory. Thus, we tested whether the intercept (population mean) of the differences in metrics differs from 0 (the expectation if the HYSPLIT and CA models produce similar results), whether this difference changes with forecast hour, whether the starting location has any effect on this difference (i.e., whether local conditions change the congruence between the CA and HYSPLIT models), and whether the elevation of the HYSPLIT model has any effect on this difference. This model allows for a non-linear relationship between the response value and the predictors. Additionally, by explicitly modelling the coordinates of the

starting position, this model considers spatial relationships between the different starting points and allows to examine whether deviations between the CA and HYSPLIT model are spatially clustered.

Validation against empirical data on FAW migration

Data on FAW migration throughout eastern Australia were contributed by Dr Melina Miles, principal entomologist at Queensland Department of Agriculture and Fisheries (QDAF). The QDAF dataset consists of the results of pheromone traps deployed for ~7 days each, starting from 25th February 2020 until the end of 2021. Traps were initially deployed in northern QLD, but rapidly became deployed in central and southern Queensland, and eventually in NSW and Victoria as FAW spread throughout the country (Figure 3).

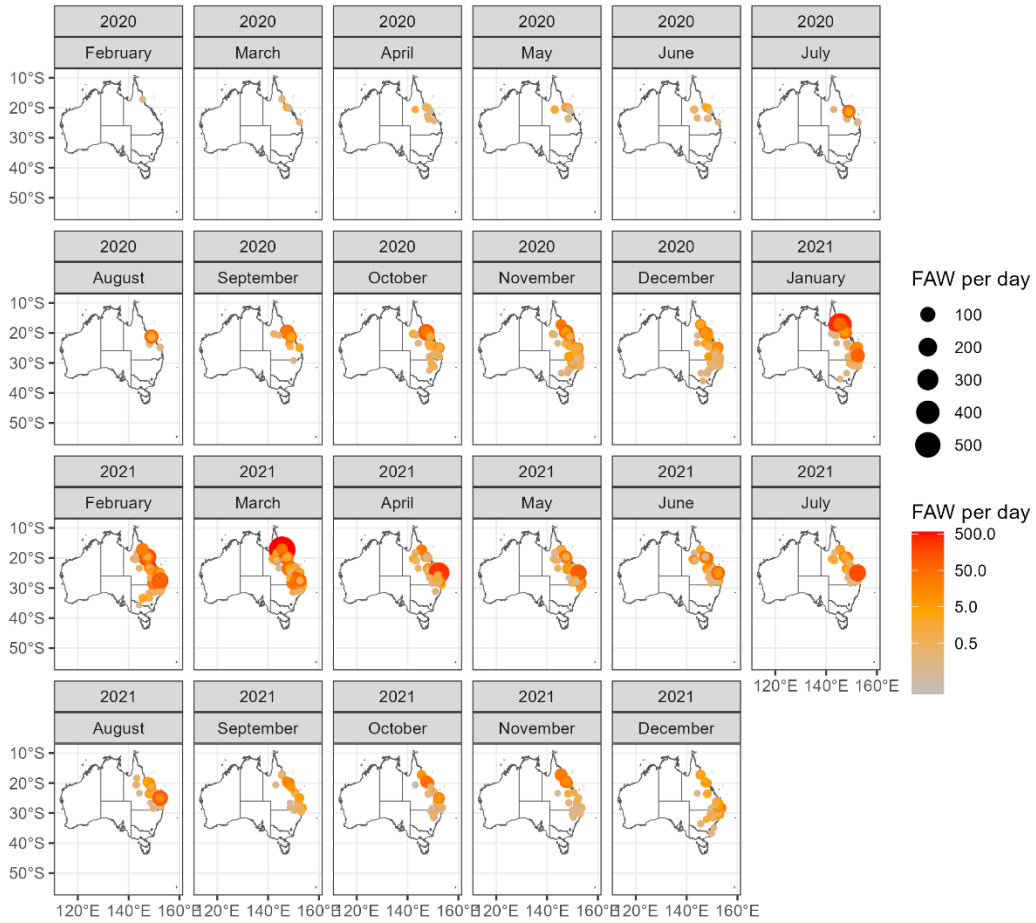


Figure 3: Maps showing the distribution of FAW throughout eastern Australia monthly, as calculated from pheromone trap data (total number of FAW trapped divided by number of days traps were open).

In order to analyse how well the CA model can predict FAW migration, we first performed spatial thinning on the trapping localities where FAW were trapped, reducing them to a single locality per 0.25 x 0.25 degree cell. We then split the dataset into 14-day intervals. We treated all cells with recorded FAW within the interval as

starting locations for simulations. We ran 10 simulations from each starting location for each day during the interval at 850 mb atmospheric level (approximately 1400 m above sea level). Starting times were sunset at each day and simulations ran for the number of hours between sunset and sunrise. Sunset and sunrise times were calculated based on Julian date, latitude, and longitude using the *photobiology* package v0.10.15 (Aphalo 2015). Simulation outputs for each interval were then summed by calculating the number of trajectories passing through each cell and converted to frequencies to generate a GIS dispersal probability layer with values ranging from 0 to 1.

We then converted the original FAW sampling results into a binary variable by treating all traps where no FAW were captured as 0 (absence) and all traps where at least one FAW was captured as 1 (presence). We thus generated a spatial presence-absence matrix for each 14-day interval. We then tested how well the dispersal probability layer for each 14-day interval can predict the presence-absence matrix of the subsequent interval. To do this, we generated the receiver operating characteristic (ROC) curve for each interval using the 'roc' function from the *pROC* package v1.18.0 (Robin et al. 2011). The ROC curve is used to assess the performance of a binary classification method with a continuous output (in our case dispersal probability), by showing the sensitivity (proportion of correctly classified positive observations; in our case presence, 1) and specificity (proportion of correctly classified negative observations; in our case absence, 0) as the output threshold (the probability value above which a cell is assigned as predicted presence) is moved across its range (0 to 1). The area under the curve (AUC; ranging from 0 to 1) is calculated using the 'auc' function from the *pROC* package for each ROC curve, with higher values representing a better classification. Traditionally, an AUC of 0.5 suggests no discrimination (i.e., dispersal probability cannot successfully predict presence or absence), 0.7 to 0.8 is considered acceptable, 0.8 to 0.9 is considered excellent performance, and >0.9 is considered outstanding performance (Hosmer and Lemeshow 2000).

We also compared the CA model's performance to three distance-based null models, by applying a distance buffer to each starting point. The buffer was created as a circle around the starting point with a radius of cells equivalent to the length (in hours) of night (see above), representing a movement of one cell per hour. The three models were:

1. Uniform: all cells within the distance buffer receive a value of 1, representing equal probability of dispersal to any point within the distance buffer.
2. Linear: the probability of dispersal in each cell *i* was calculated as:
   $p = (d_{max} - d_i) / d_{max}$
   where d_i is the distance from cell *i* to the starting cell, and d_{max} is the maximum distance. This represents a probability of dispersal that decreases linearly with distance from the starting point.
3. Exponential: the probability of dispersal in each cell *i* was calculated as:
   $p = e^{-di} / e^{-dmax}$

where d_i is the distance from cell i to the starting cell, and d_{max} is the maximum distance. This represents a probability of dispersal that decreases exponentially with distance from the starting point.

ROC curves were generated and AUC values calculated for the null models as described above for the CA model. We then analysed whether AUC values change with time and between CA and the null models by fitting a GAM with a beta regression family using the *mgcv* package with the following formula:

$AUC_i \sim s(N_i, by = M)$

Where AUC_i is the AUC of interval i, N_i is the sequential number of the interval i, and M is a factor describing the model under examination used in factor smooth interactions, allowing different smooth parameters for different levels of the factor.

Results

Comparison with the NOAA HYSPLIT simulations

Overall, our validation efforts suggest that congruence between HYSPLIT and CA model trajectories is relatively low and decreases over longer forecast hours.

When comparing HYPSLIT and CA, the null hypothesis (no difference) was rejected more often than expected by chance for all timesteps for distance (p value of binomial exact test < 0.01 for all timesteps), suggesting CA and HYSPLIT models differ in travelled distances. Earlier during forecasts, the congruence between CA and HYSPLIT is higher for azimuth, and the binomial exact tests suggest angles of travel do not differ much between CA and HYSPLIT for the first four timesteps (null not rejected more often than it wasn't; $p > 0.6$). From the fifth timestep onwards, all binomial exact tests suggest that azimuth significantly differs between CA and HYSPLIT ($p \leq 0.01$), and congruence diminished to similar, albeit slightly higher, levels as for distance (Figure 4; Table 1).

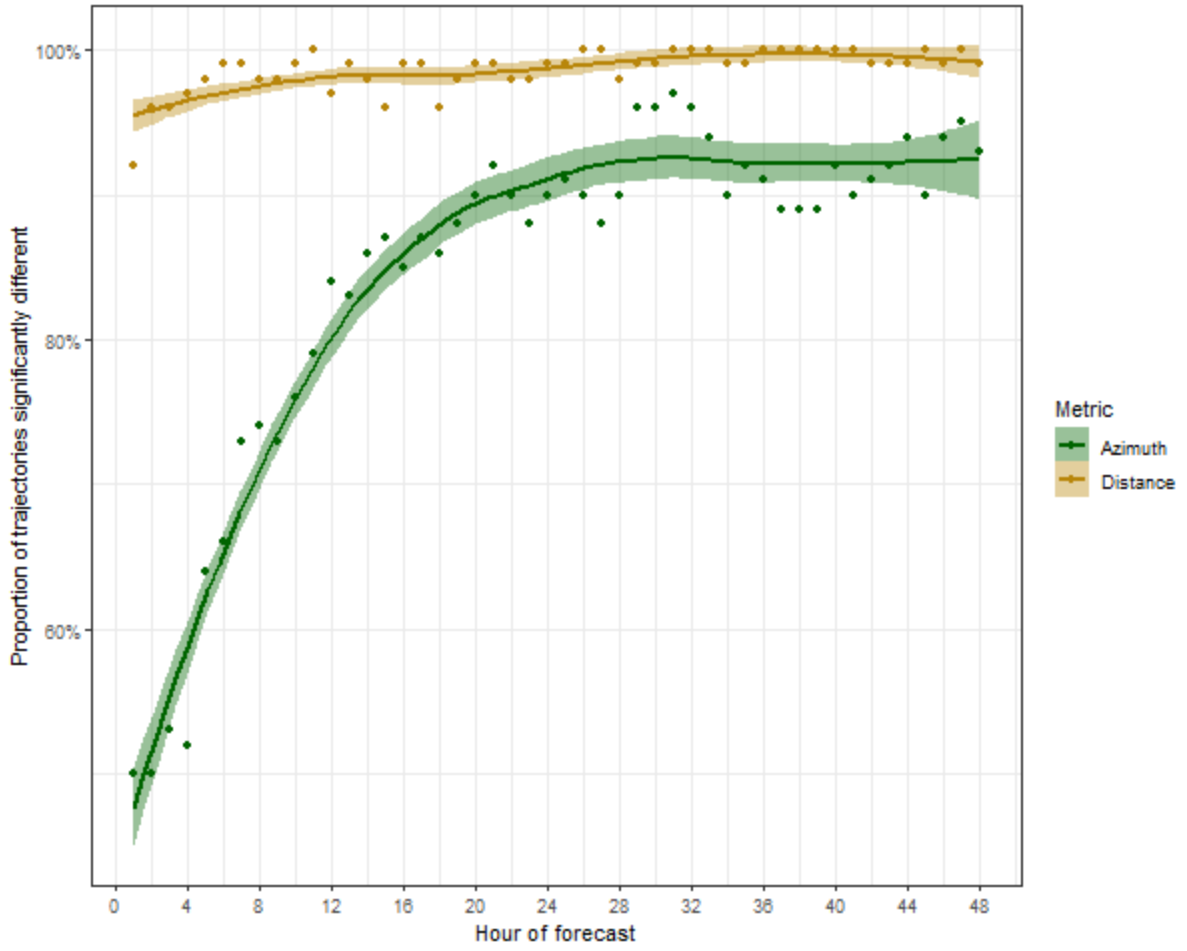


Figure 4: Proportion of trajectories, out of 100 random starting points, in which the null hypothesis (the distribution of CA trajectory metrics sampled from a population with a mean equal to the HYSPLIT trajectory) was rejected, at each timestep of forecast.

Table 1: Summary of comparisons of HYSPLIT and CA simulations. The % Distance and % Azimuth columns list the percentage of trajectories, out of 100 random starting points, in which the null hypothesis (the distribution of CA trajectory metrics sampled from a population with a mean equal to the HYSPLIT trajectory) was rejected in a one sample t-test. The *p* Distance and *p* Azimuth columns list the *p*-values of associated binomial exact tests to assess whether the proportion significantly differs from 0.5.

Forecast hour	% Distance	*p* Distance	% Azimuth	*p* Azimuth
1	92%	0.00	50%	1.00
2	96%	0.00	50%	1.00
3	96%	0.00	53%	0.62
4	97%	0.00	52%	0.76

5	98%	0.00	64%	0.01
6	99%	0.00	66%	0.00
7	99%	0.00	73%	0.00
8	98%	0.00	74%	0.00
9	98%	0.00	73%	0.00
10	99%	0.00	76%	0.00
11	100%	0.00	79%	0.00
12	97%	0.00	84%	0.00
13	99%	0.00	83%	0.00
14	98%	0.00	86%	0.00
15	96%	0.00	87%	0.00
16	99%	0.00	85%	0.00
17	99%	0.00	87%	0.00
18	96%	0.00	86%	0.00
19	98%	0.00	88%	0.00
20	99%	0.00	90%	0.00
21	99%	0.00	92%	0.00
22	98%	0.00	90%	0.00
23	98%	0.00	88%	0.00
24	99%	0.00	90%	0.00
25	99%	0.00	91%	0.00
26	100%	0.00	90%	0.00
27	100%	0.00	88%	0.00
28	98%	0.00	90%	0.00
29	99%	0.00	96%	0.00

30	99%	0.00	96%	0.00
31	100%	0.00	97%	0.00
32	100%	0.00	96%	0.00
33	100%	0.00	94%	0.00
34	99%	0.00	90%	0.00
35	99%	0.00	92%	0.00
36	100%	0.00	91%	0.00
37	100%	0.00	89%	0.00
38	100%	0.00	89%	0.00
39	100%	0.00	89%	0.00
40	100%	0.00	92%	0.00
41	100%	0.00	90%	0.00
42	99%	0.00	91%	0.00
43	99%	0.00	92%	0.00
44	99%	0.00	94%	0.00
45	100%	0.00	90%	0.00
46	99%	0.00	94%	0.00
47	100%	0.00	95%	0.00
48	99%	0.00	93%	0.00

The fitted GAMs (Table 2) suggest that the difference in distance is centred around 0, at least in early hours of forecast, but large divergences in predicted distance travelled (> 500 km) occur after ~12 hours of forecast (Figure 5A). These divergences are also biased towards a negative difference between CA and HYSAPLIT distances, meaning that, on average and after taking spatial patterns into account, CA predicts shorter distances travelled than HYSPLIT. For offset in angle, there is large variation in offset for all forecast hours, with most points between 0^{o} and 90^{o}. Thus, the fitted GAMs suggest a consistent predicted offset of ~23-57^{o} in the HYSPLIT trajectory

compared to the CA trajectory, meaning HYSPLIT trajectories are consistently somewhat clockwise compared to CA trajectories (Figure 5B).

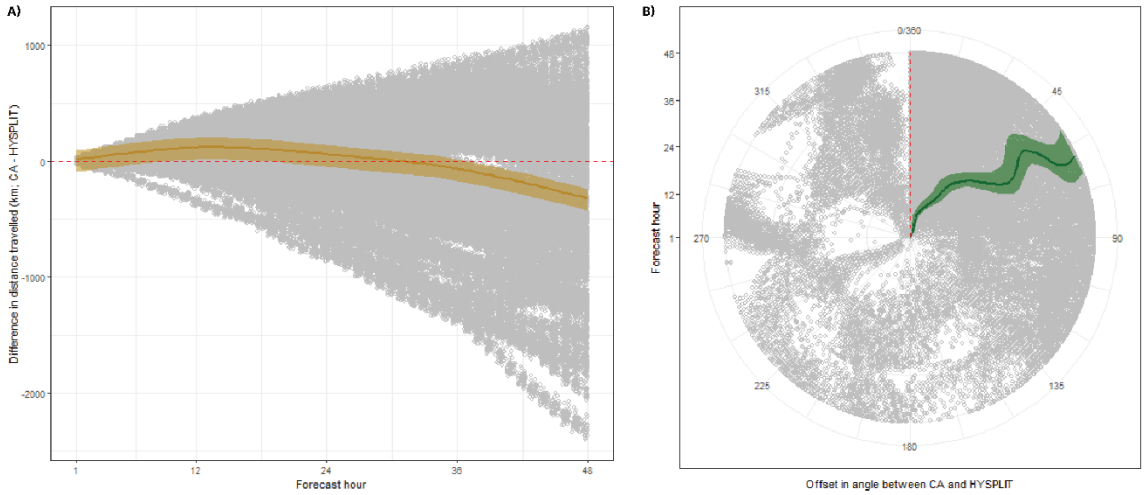


Figure 5: Observations (grey points) and model predictions (gold and green coloured lines, with confidence intervals in lighter shades) of (A) difference in distance travelled (gold), and (B) offset in angle (green) between HYSPLIT and CA models as a function of forecast hour.

The GAMs suggest spatial clustering in the effects on distance differences and angle offset ($p < 0.01$ and $p = 0.03$, respectively). Some regions "pull" the differences in distances in a shorter or longer direction, and some "pull" the offset in angle in a more clockwise or anti-clockwise direction, but there is overall weak congruence between these regions. This is likely due to spatial autocorrelation in wind regimes and might suggest that the accuracy of model predictions can be spatially dependent on the starting location of a simulation.

The elevation of the HYSPLIT model has a significant effect on difference in distance travelled ($p < 0.01$) but not on offset in angle ($p = 0.79$). A closer inspection of the partial effects plot of elevation on the two different metrics shows that the differences are minimised around elevations close to 500 m (Figure 6), roughly equivalent to the 950 mb atmospheric level at which CA simulations were run. This would suggest that a major reason for the lack of congruence between HYSPLIT and CA could be that the HYSPLIT model includes vertical movements of particles; that is, particles can be exposed to differing wind regimes as they change in elevation. Conversely, the CA model makes predictions at a constant elevation, carrying the implicit assumption that there is no vertical movement. This difference in vertical movement being a strong driver of the difference in model predictions is supported by the predicted differences in azimuth increasing with forecast hour (Figures 4 & 5B), as the longer the simulations run for, the more likely they are to diverge in altitude (Figure 7).

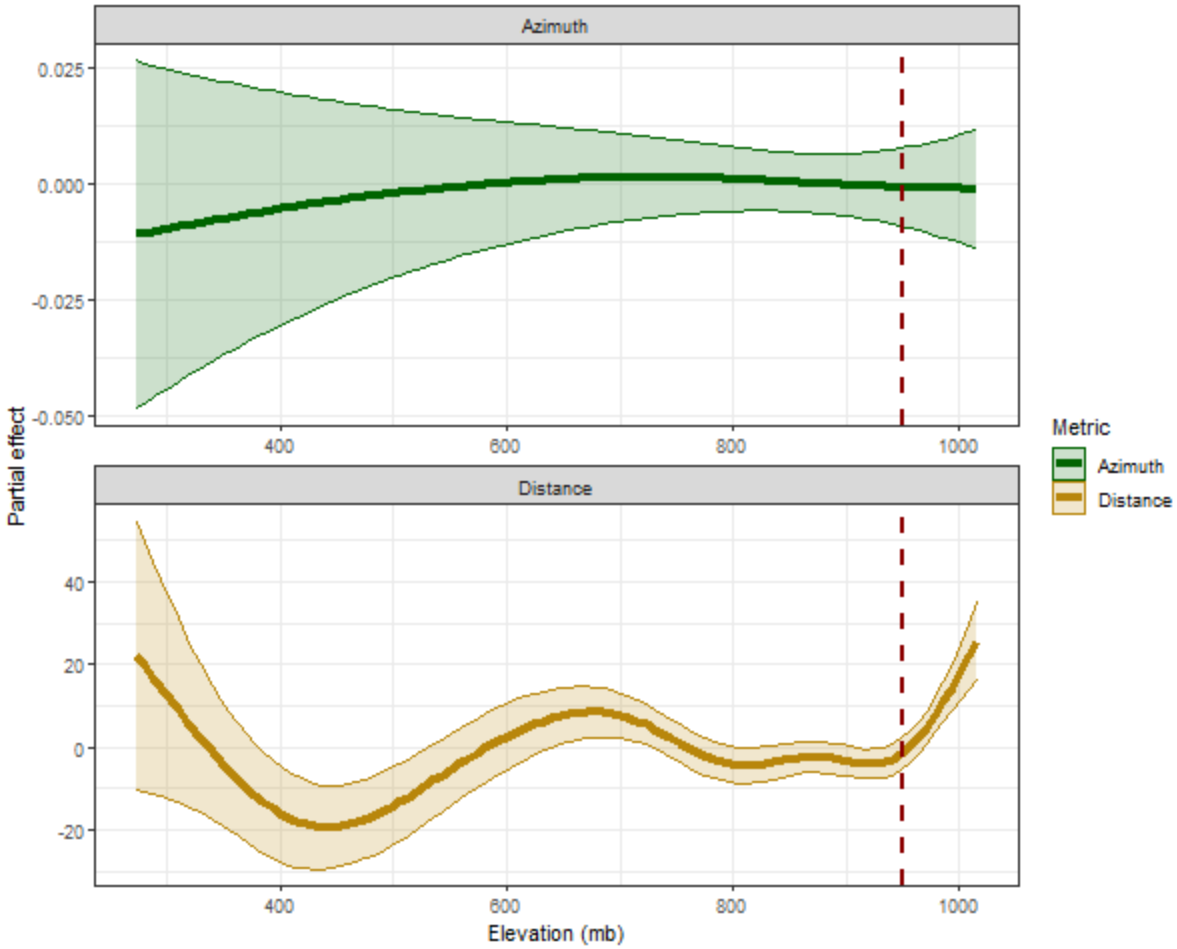


Figure 6: Partial effects of elevation of HYSPLIT simulation (in mb) on difference in distance travelled (gold) and offset in angle (green). The dashed red line represents 950 mb, the elevation used for the CA simulations.

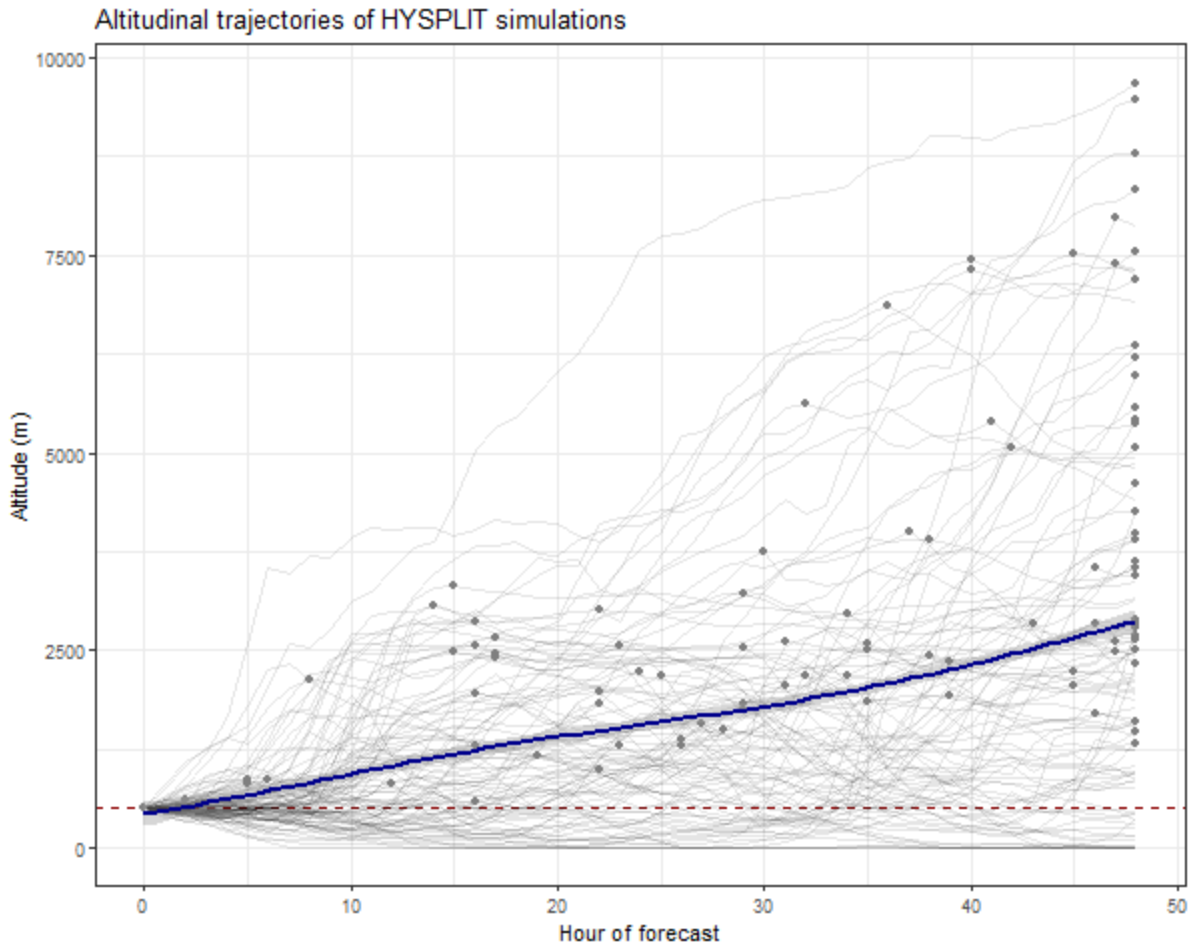


Figure 7: Altitudinal trajectories of all simulated HYSPLIT trajectories. Grey lines represent individual simulated trajectories, with grey dots representing the maximum altitude reached by each trajectory. The blue line represents the average across all 100 simulated trajectories. The dashed red line represents 500 m, the starting point for all simulations and roughly equivalent to 950 mb, which was set as the elevation for all CA simulations.

Table 2: Summary of GAMs comparing differences in distance travelled and offset in angle (azimuth) between HYSPLIT and CA simulations. The EDF column lists the effective degrees of freedom for each predictor, a measure of non-linearity (EDF = 1 equivalent to linear relationship, EDF > 1 and ≤ 2 weakly non-linear relationship, EDF > 2 highly non-linear relationship; Zuur et al. 2009), the *p* column lists the *p*-value for the predictor, and the R^2 column lists the R^2 of the entire model.

Metric	Predictor	EDF	*p*	R^2
Distance	Hour of forecast	8.06	< 0.01	0.083
	Spatial coordinates	27.42	< 0.01	
	Elevation	7.17	< 0.01	

Azimuth	Hour of forecast	8.91	< 0.01	0.037
	Spatial coordinates	17.33	0.03	
	Elevation	1.52	0.79	

Whether the CA assumption of vertical stability is valid for modelling FAW movements is currently unknown and depends on FAW capacity to shift their vertical position in the air column during migration flight. However, several HYSPLIT trajectories include particles travelling to altitudes well over 5000 m (Figure 7), and these may certainly be biologically unrealistic for FAW movement due to low temperatures inhibiting flight, meaning that HYSPLIT's implicit assumption of complete vertical mobility depending on wind regimes is also untested and may not be valid.

Several directions forward present themselves: predictions with the CA model can be limited to relatively short forecast periods (e.g., up to ~12h) to limit deviation from HYSPLIT predictions. Alternatively, attempts can be made to include vertical movements into the CA modelling framework based on vertical components of wind, although that may come at the cost of computational efficiency and carries the implicit assumption of HYSPLIT models of complete vertical mobility. A promising approach is to simulate CA trajectories for several different constant flight elevations, spanning a biologically plausible range (e.g., Wu et al. 2022). Then, model predictions can be summed over the different elevations, representing relative probabilities of dispersal over a range of flight altitudes and trajectories. However, this approach would greatly compound the number of simulations necessary to make predictions, and so would require adapting the CA simulation code to a faster programming environment to be practical.

Validation against empirical data on FAW migration

Across all models (CA and three null distance-based models; uniform, linear and exponential), predictive capability as measured by AUC is relatively high (mean AUC per model 0.70-0.92) when predicting FAW presence between February to October 2020, after which model AUC drops considerably (mean AUC per model 0.52-0.70). However, the models also differ from each other substantially in initial mean AUC and in the degree of drop-off (Figure 8). The fitted GAM model supports different smooth parameters for each model (Table 3).

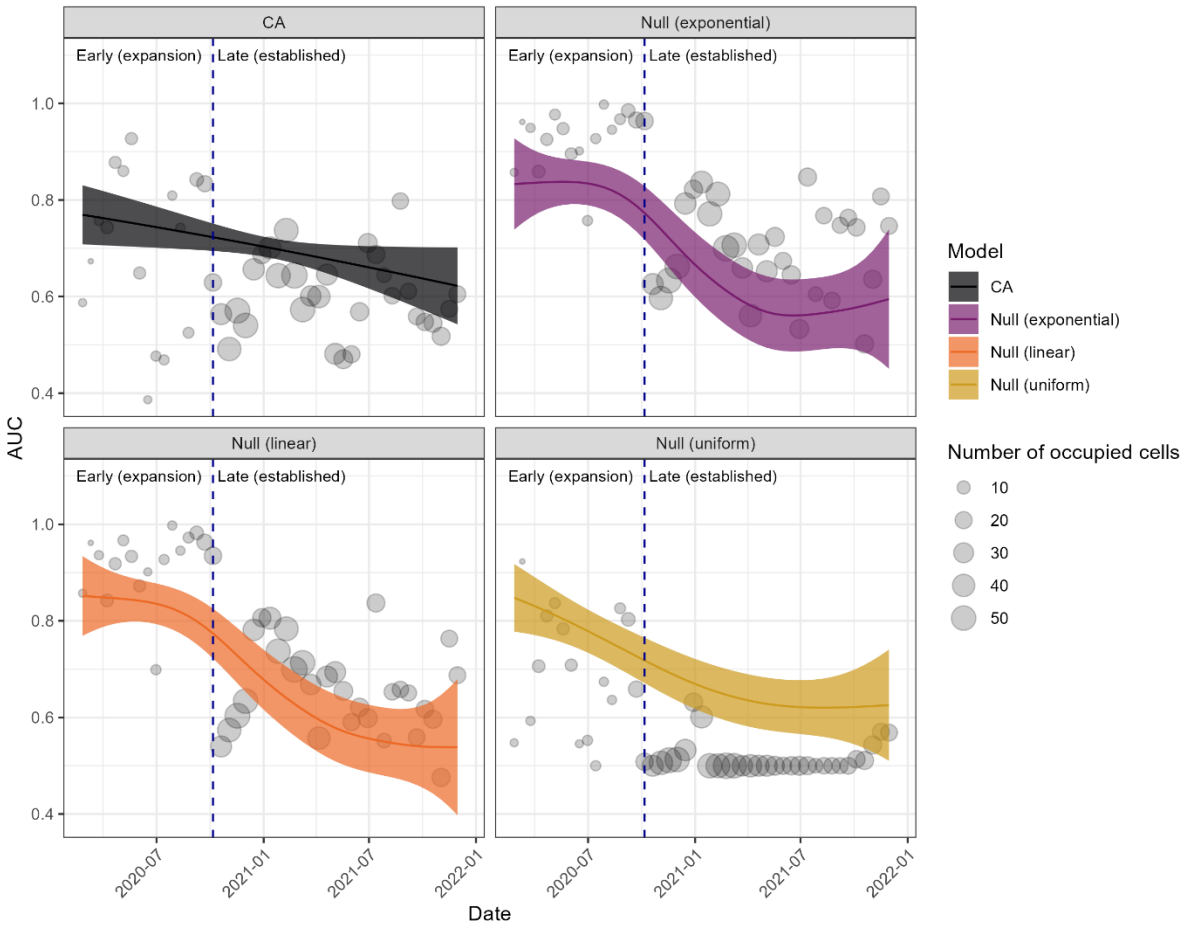


Figure 8: Scatterplots showing the distribution of AUC values per dispersal model per date interval, with the size of points corresponding to the number of occupied cells in the raster (presence points). The coloured lines and shaded areas represent the predictions and 95% CI from the fitted GAM model. The dashed blue vertical line represents the start of October 2020, the cut-off between the early (expansion) period and the late (established) period.

Table 3: Summary of GAM comparing AUC as a function of date interval under different dispersal models. The EDF column lists the effective degrees of freedom for each smooth parameter, a measure of non-linearity (EDF = 1 equivalent to linear relationship, EDF > 1 and ≤ 2 weakly non-linear relationship, EDF > 2 highly non-linear relationship; Zuur et al. 2009), the *p* column lists the *p*-value for the smooth parameter, and the R^2 column lists the R^2 of the entire model.

Model	EDF	*p*	R^2
CA	1.00	0.03	0.245
Null (uniform)	2.06	< 0.01	

Null (linear)	2.82	< 0.01
Null (exponential)	3.15	< 0.01

The cutoff period of October 2020 seems to correspond well to a shift in FAW dynamics from few occupied localities to many occupied localities, reflected both in the number of presence points but also in the proportion of traps with successful FAW captures (Figure 9). These two periods, which we term here 'early (expansion)' and 'late (established)', appear to correspond well to the theoretical expectations from invasion biology (Allendorf and Lundquist 2003; Ricciardi 2012), wherein an invasive species can experience a lag period between initial establishment and exponential population growth (Crooks and Soulé 1999; Crooks 2005). Our models would suggest that long-distance dispersal has a much stronger predictive capability in this initial stage, presumably due to local demographic effects and short-distance dispersal becoming more important once the population is established.

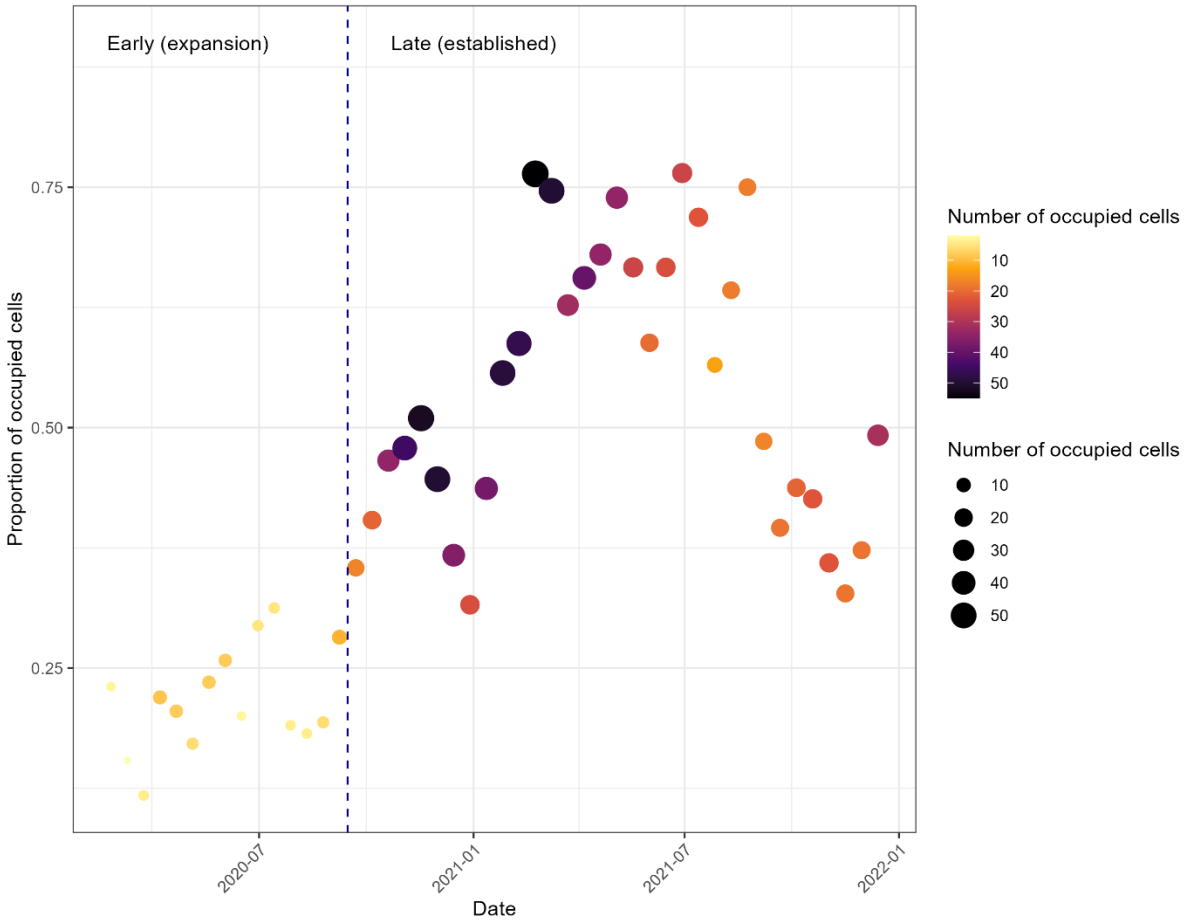
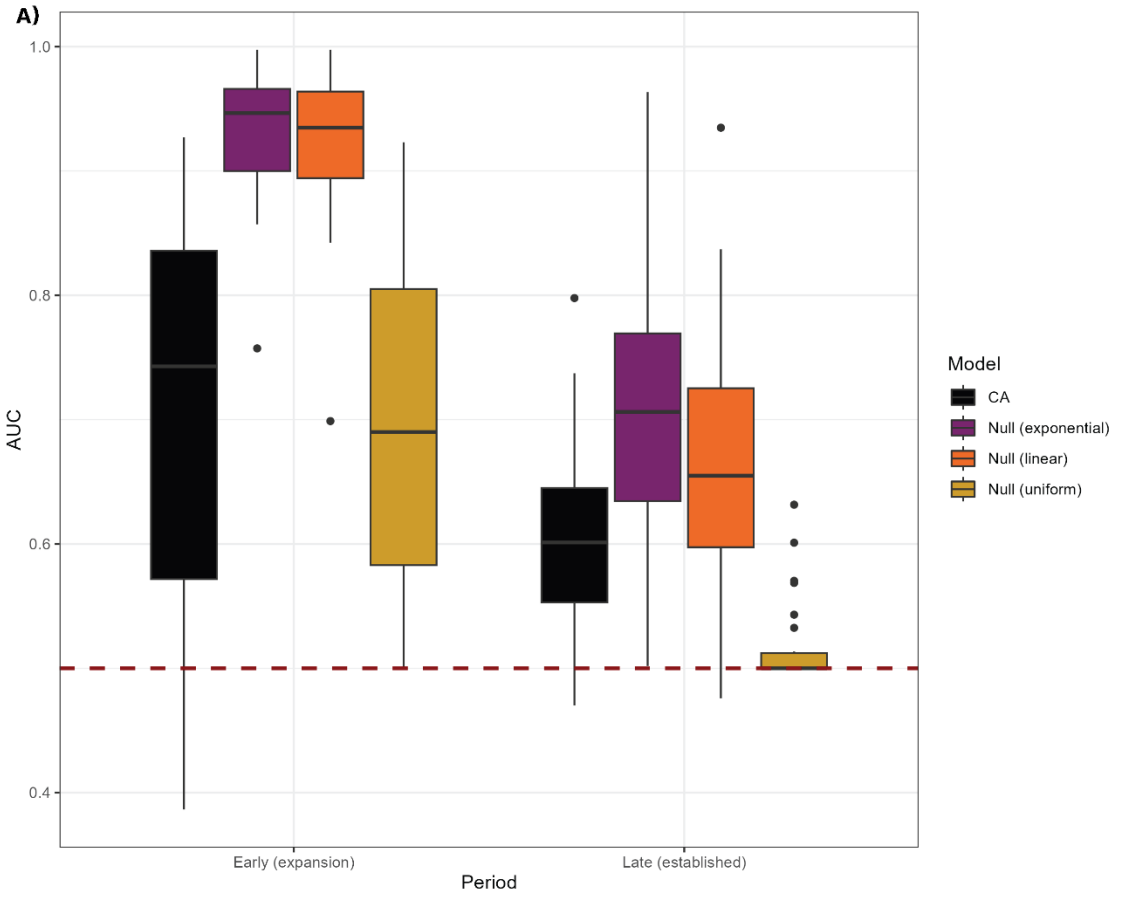


Figure 9: Scatterplot showing the temporal dynamics of the FAW spread through eastern Australia. The y-axis captures the proportion of occupied cells in the raster out of all cells where traps were deployed. The size and colour of the dots corresponds to the absolute number of occupied cells. The dashed blue line corresponds to the start of October 2020,

where we see a shift from low proportion and number of occupied cells (early period) to mid-high proportion and high number of occupied cells (late period).

We ran a two-factor analysis of variance (ANOVA) to examine whether dispersal models differ in their AUC scores between each other within and among the two periods. In both periods, the distance-based linear and exponential dispersal models clearly outperform the CA model, and both have outstanding predictive performance (AUC > 0.9; Figure 10). Meanwhile, the CA model has adequate performance in the early period that is not statistically distinguishable from the uniform null model (p = 1.00), and poor performance in the late period. However, the uniform null model performs worst in the late period and is no better than random (Figures 8 & 10).


A)
1.0
0.8
0.6
0.4
AUC
Early (expansion)
Late (established)
Period
Model
CA
Null (exponential)
Null (linear)
Null (uniform)

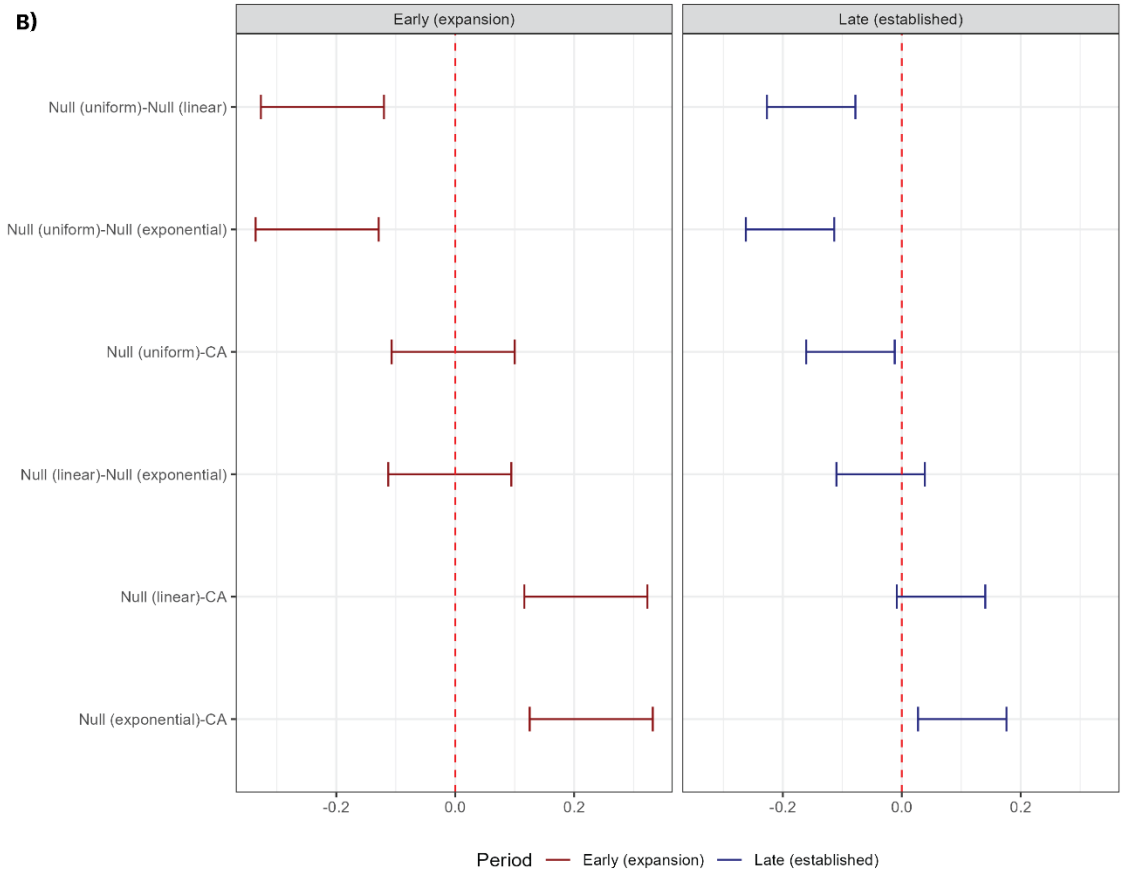

B)
Early (expansion)
Late (established)
Null (uniform)-Null (linear)
Null (uniform)-Null (exponential)
Null (uniform)-CA
Null (linear)-Null (exponential)
Null (linear)-CA
Null (exponential)-CA
-0.2
0.0
0.2
Period
Early (expansion)
Late (established)

Figure 10: Comparison of model performance for CA and three null dispersal models. (A) Boxplots showing the distribution of AUC values of the four models in the two time periods. The dashed horizontal red line represents an AUC of 0.5 – equivalent to random classification. (B) Results of a post-hoc Tukey test comparing the mean AUC values of the different models in the two time periods. The dashed vertical red lines represent a difference of 0 in means. The whiskers represent the upper and lower confidence intervals of the estimate of differences in the examined pair of models – whiskers that overlap 0 mean there is no statistically significant difference between the two means.

The relatively poor performance of the CA model compared to the distance based linear and exponential models is intriguing and poses questions that will require future research. Three possible reasons appear most plausible:

1. The model makes several unrealistic biological assumptions due to lack of relevant data. It is established that noctuid moths engage in nocturnal migration beginning at dusk at elevations of ~500 m (Wood et al. 2009) and their migration flights are strongly assisted by tailwinds (Alerstam et al. 2011). However, it is also known that noctuid moths are strongly selective of winds blowing towards preferred directions (Alerstam et al. 2011) and very little is known about FAW selective behaviour regarding migration cues and duration of nightly flights. More data on the biology of FAW migratory behaviour will help to create more realistic and plausible parameters for model simulations and could increase predictive performance.
2. The empirical dataset used for validation, while extensive, was not sampled systematically in a method designed to validate a dispersal model. Localities were not sampled sequentially in a constant manner. Therefore, the dataset might not be fully suited to validate the dispersal model.
3. Spatial spread of FAW is a result of both long-distance and short-distance dispersal, the latter of which is presumably less contingent on wind movements at high elevations. Therefore, the distance-based dispersal models, which are agnostic to direction, might be better suited to capture these movements.

Web tool

We have developed a web application for growers and agronomists to easily access and interact with the wind-dispersal model. This web application was developed with R Shiny (Chang et al. 2022) and is currently hosted on Cesar Australia's web servers (Figure 11). The application is currently being reviewed by stakeholders and will be made publicly available on Cesar Australia's website pending final approval.

The application automatically sources new weather forecasts from GFS daily at 9:00 AM Australian Eastern Standard Time (shortly after the release of the equivalent 18:00 UTC forecast cycle). Forecast data are stored for a week before being cleared from the memory cache.

The application can be used in two ways:

1. Forecast dispersal from a location with a known FAW infestation: this can be used to give likely movements of FAW for up to 48 hours and can serve as an early warning system for agronomists and growers that FAW may arrive at their paddock soon.
2. Hindcast dispersal from a location with a novel FAW infestation: this can be used to investigate where the new infestation originated from and can help in tracking FAW movements and spread.

The application was designed to allow end-users with no coding experience to access and interact with the model. Users can select a location in Australia to run simulations – this can be done either by inputting latitude and longitude, or by clicking on an interactive map. Simulations can then be run to either forecast or hindcast dispersal – in the former the selected locality will serve as a starting point, and in the latter as the end point. Users can determine the start time (for a forecast) or end time (for a hindcast) of the simulation converted to the time zone of their choice. Users then choose the length of the simulation in hours. Ten simulations are then run at an atmospheric level of 950 mb (approximately 500 m above sea level). An output map is generated showing the frequency of simulated trajectories. The map includes a colour scale, and so can be easily interpreted as the predicted probability of dispersal pathways.

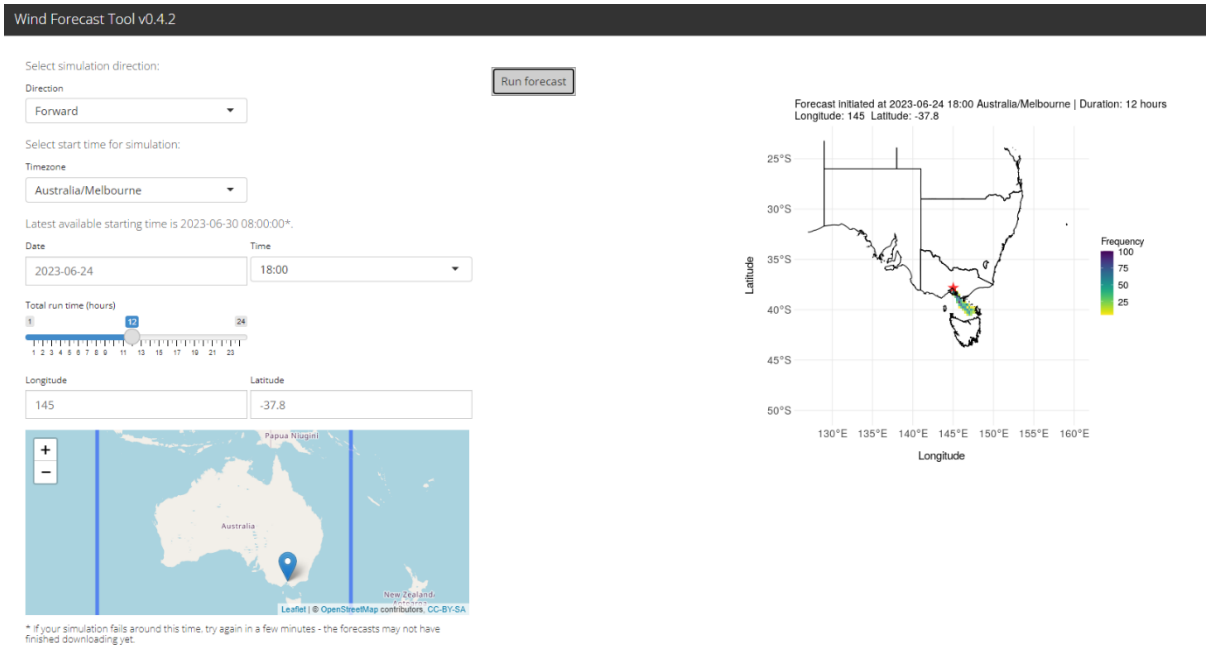


Figure 11: Version 0.4.2 of the Wind Forecast Tool web application. The user inputs (on the left) consist of the direction of simulation (forward/backward), timezone, time and starting location of the simulation, and duration of simulation. The app then runs 10 simulations using the input parameters and generates the output (on the right): a map showing frequencies of simulated trajectories.

Conclusion

We have developed a novel CA model of wind-assisted dispersal for FAW, which may be easily accessed and interacted with through a user-friendly web application to fore- and hindcast FAW movements. This has implications both for early-warning biosecurity surveillance, as well as for general research into invasion pathways.

Some limitations for widespread use of the model are still evident. Our comparison efforts show that the CA model produces slightly different results from the industry standard physical HYSPLIT model. These differences appear to stem mostly from vertical mobility – HYSPLIT allows trajectories to change in elevation based on vertical wind components, whereas our CA model operates at a set and predetermined atmospheric layer. This assumption may be relaxed somewhat in future extensions of the CA model – however, to which extent remains an open question. With both HYSPLIT and CA, a biologically realistic upper elevation bound should be set to reflect physiological limits beyond which FAW cannot survive. Beyond that, the degree to which FAW can actively control their vertical position in the air column, and not just be passively carried by winds, needs to be determined.

Similarly, validation of the model against empirical data revealed that a simple distance-based dispersal model can often outperform the spatially explicit CA model in predicting FAW occurrence. Whether this is due to limitations of the model itself, or whether wind-assisted long-distance dispersal is not an important contributor to spatial spread of FAW remains to be determined. The silver lining is that relatively simple distance-based long-distance dispersal models, such as the one used in FAW demographic modelling by Maino and colleagues (2021), can be adequate to model spatially explicit population dynamics, at least during the establishment stage of a biological invasion – or at the leading edge of an invasion front.

Further research into FAW migration in Australia will be necessary to uncover migration cues, movement biology, and flight behaviour. These data will be important to improve dispersal models by incorporating biologically plausible parameters, and to adequately validate the performance of dispersal models. Regardless, our model can still prove extremely useful to growers and agronomists. By incorporating our model into an easy-to-use web application, we have equipped users with a simple tool to receive early warnings of possible FAW incursions. While the model cannot yet predict FAW presence with certainty, it can provide a warning that conditions are suitable for FAW. This is an important step towards a more comprehensive surveillance and mitigation strategy for this economically important pest species.

Acknowledgments

We are extremely grateful to Dr Melina Miles (QDAF) for sharing empirical data on FAW movements in Australia. Dr James Maino (Cesar Australia) and Dr Roozbeh Valavi (CSIRO) played a key role in the early stages of the CA model development and on the

design of the web application – we are thankful for their support throughout the project. We are grateful to Dr Lizzy Lowe (Cesar Australia) for her assistance in designing and testing the web application.

References

Alerstam T, Chapman JW, Bäkman J, Smith AD, Karlsson H, Nilsson C, Reynolds DR, Klaassen RHG, Hill JK. 2011. Convergent patterns of long-distance nocturnal migration in noctuid moths and passerine birds. *Proceedings of the Royal Society B*, 278(1721), 3074–3080.

Allendorf FW, Lundquist LL. 2003. Introduction: population biology, evolution, and control of invasive species. *Conservation Biology*, 17(1), 24–30.

Aphalo PJ. 2015. The r4photobiology suite: spectral irradiance. *UV4Plants Bulletin*, 2015(1), 21–29.

Chang W, Cheng J, Allaire J, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dipert A, Borges B. 2022. *shiny: Web Application Framework for R*. R package version 1.7.4, https://CRAN.R-project.org/package=shiny.

Crooks JA. 2005. Lag times and exotic species: the ecology and management of biological invasions in slow-motion. *Écoscience*, 12(3), 316–329.

Crooks JA, Soulé ME. 1999. Lag times in population explosions of invasive species: causes and implications. In: *Invasive Species and Biodiversity Management* (eds. Sandlund OT, Schei PJ, Viken A) 103–125. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Drake VA. 1994. The influence of weather and climate on agriculturally important insects: an Australian perspective. *Australian Journal of Agricultural Research*, 45(3), 487–509.

Drake VA, Farrow RA. 1988. The influence of atmospheric structure and motions on insect migration. *Annual Review of Entomology*, 33, 183–210.

Draxler RR, Hess GD. 1998. An overview of the HYSPLIT_4 modelling system for trajectories. *Australian Meteorological Magazine*, 47, 295–308.

Gregg PC, Del Socorro AP, Rochester WA. 2001. Field test of a model of migration of moths (Lepidoptera: Noctuidae) in inland Australia. *Australian Journal of Entomology*, 40(3), 249–256.

Hosmer DW, Lemeshow S. 2000. *Applied Logistic Regression*, 2nd Ed. 160-164. John Wiley and Sons, New York, USA.

Maino JL, Schouten R, Overton K, Day R, Ekesi S, Bett B, Barton M, Gregg PC, Umina PA, Reynolds OL. 2021. Regional and seasonal activity predictions for fall armyworm in Australia. *Current Research in Insect Science*, 1, 100010.

Mitchell ER, McNeil JN, Westbrook JK, Silvain J-F, Lalanne-Cassou B, Chalfant RB, Pair SD, Waddill VH, Sotomayor-Rios A, Proshold FI. 1991. Seasonal periodicity of fall

armyworm (Lepidoptera: Noctuidae) in the Caribbean basin and northward to Canada. *Journal of Entomological Science*, 26(1), 39–50.

R Core Team. 2022. *R: A language and environment for statistical computing*. URL: https://www.R-project.org/. R Foundation for Statistical Computing, Vienna, Austria.

Ricciardi A. 2012. Invasive species. In: *Ecological systems: selected entries from the Encyclopedia of sustainability science and technology* (ed. Meyers RA) 161–178. Springer, New York, USA.

Robin X, Turck N, Hainard A, Tiberti N, Lisacek F, Sanchez J-C, Müller M. 2011. pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, 12, 77.

Rolph GD, Stein AF, Stunder BJB. 2017. Real-time Environmental Applications and Display sYstem: READY. *Environmental Modelling & Software*, 95, 210–228.

Stein AF, Draxler RR, Rolph GD, Stunder BJB, Cohen MD, Ngan F. 2015. NOAA's HYSPLIT atmospheric transport and dispersion modeling system. *Bulletin of the American Meteorological Society*, 96(12), 2059–2077.

Westbrook JK. 2008. Noctuid migration in Texas within the nocturnal aeroecological boundary layer. *Integrative and Comparative Biology*, 48(1), 99–106.

Westbrook JK, Fleischer SJ, Jairam S, Meagher RL, Nagoshi RN. 2019. Multigenerational migration of fall armyworm, a pest insect. *Ecosphere*, 10(11), e02919.

Westbrook JK, Nagoshi RN, Meagher RL, Fleischer SJ, Jairam S. 2016. Modeling seasonal migration of fall armyworm moths. *International Journal of Biometeorology*, 60(2), 255–267.

Westbrook JK, Sparks, AN. 1986. The role of atmospheric transport in the economic fall armyworm (Lepidoptera: Noctuidae) infestations in the southeastern United States in 1977. *Florida Entomologist*, 69(3), 492–502.

Wolfram S. 1984. Cellular automata as models of complexity. *Nature*, 311, 419–424.

Wood CR, Reynolds DR, Wells PM, Barlow JF, Woiwood IP, Chapman JW. 2009. *Bulletin of entomological research*, 99(5), 525–535.

Wood SN. 2003. Thin-plate regression splines. *Journal of the Royal Statistical Society B*, 65(1), 95–114.

Wood SN. 2011. Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society B*, 73(1), 3–36.

Wu M-F, Qi G-J, Chen H, Ma J, Liu J, Jiang Y-Y, Lee G-S, Otuka A, Hu G. 2022. Overseas immigration of fall armyworm, *Spodoptera frugiperda* (Lepidoptera: Noctuidae), invading Korea and Japan in 2019. *Insect Science*, 29(2), 505–520.

Zuur AF, Ieno EN, Walker N, Saveliev AA, Smith GM. 2009. *Mixed effects models and extensions in ecology with R*. Springer, New York, USA.

Appendix 1. R code to run CA wind-assisted dispersal model

Background

The fall armyworm (FAW, *Spodoptera frugiperda*) is a lepidopteran pest recently established in Australia. It feeds in large numbers on the leaves, stems and reproductive parts of more than 350 plant species, causing major damage to economically important crops.

The CA wind-assisted dispersal model for the FAW was developed under Cesar Australia project CE1805CR3, with funding by Plant Health Australia. It uses gridded weather forecast data on wind speed and direction to calculate spatial trajectories of FAW based on user-provided input parameters.

Model structure

The following helper functions are used to extract wind U and V components from weather data, to calculate wind speed and direction from U and V components, and to convert calculated directions to cells in a grid.

```
# extract the component from names
gfs_names <- function(path = "Data/") {
  require(dplyr)
  require(purrr)

  files <- list.files(path, pattern = "^gfs_", recursive = FALSE)

  gfs_names <- strsplit(files, "_") %>%
    map(function(x) {
      matrix(x, ncol = 6, byrow = FALSE) %>%
        as.data.frame() %>%
        setNames(c("gfs", "comp", "level", "date", "start", "forecast")) %>%
        dplyr::select(-gfs)
    }) %>%
    do.call(rbind.data.frame, .) %>%
    mutate(file = files) %>%
    relocate(file)

  return(gfs_names)
}

wind_direction <- function(u, v) {
  require(terra)
  at <- 180 + (atan2(u, v) * 180 / pi)
  ad <- terra::app(at, function(x)
    x %% 360)
  return(ad)
}

wind_speed <- function(u, v) {
  require(terra)
  r <- sqrt(v * v + u * u)
```

```
  return(r)
}

# read the u component
read_u <-
  function(path = "Data/",
           files_list,
           fcast,
           lev = "850mb") {
    files_list %>%
      dplyr::filter(comp == "ugrd") %>%
      dplyr::filter(level == lev) %>%
      dplyr::filter(forecast == fcast) %>%
      pull(file) %>%
      file.path(path, .) %>%
      terra::rast()
  }
# read the u component
read_v <-
  function(path = "Data/",
           files_list,
           fcast,
           lev = "850mb") {
    files_list %>%
      dplyr::filter(comp == "vgrd") %>%
      dplyr::filter(level == lev) %>%
      dplyr::filter(forecast == fcast) %>%
      pull(file) %>%
      file.path(path, .) %>%
      terra::rast()
  }

next_cell <- function(x, i, j) {
  inc <- 1
  # define the direction
  if (x <= 22.5 || x > 337.5) {
    ## N
    return(c(i, j + inc))
  } else if (x > 22.5 && x <= 67.5) {
    ## NE
    return(c(i - inc, j + inc))
  } else if (x > 67.5 && x <= 112.5) {
    ## E
    return(c(i - inc, j))
  } else if (x > 112.5 && x <= 157.5) {
    ## SE
    return(c(i - inc, j - inc))
  } else if (x > 157.5 && x <= 202.5) {
    ## S
    return(c(i, j - inc))
  } else if (x > 202.5 && x <= 247.5) {
    ## SW
    return(c(i + inc, j - inc))
  } else if (x > 247.5 && x <= 292.5) {
    ## W
    return(c(i + inc, j))
  } else if (x > 292.5 && x <= 337.5) {
```

```
    ## NW
    return(c(i + inc, j + inc))
  }
}
```

The model itself is run with a single function. It sources weather forecast data from the selected filepath and runs simulations with the input specifications. Simulations can be run from multiple localities, can be forecast or hindcast, and can be run in parallel. The parallel application is highly recommended if simulating multiple starting locations.

```
# simulate wind dispersal with cellular automata
wind_sim <-
  function(data_path = "wind-data",
           # filepath to weather data
           coords = list(),
           # list of starting coordinates, each a vector of c(longitude, latitude)
           nforecast = 24,
           # number of forecast hours
           nsim = 10,
           # number of simulations to calculate frequency
           fdate = "20220524",
           # the forecast data
           fhour = "18",
           # the forecast hour
           atm_level = "850mb",
           cellsize = 25000,
           full = F,
           # if TRUE, generate a dataframe of endpoints per time step
           parallel = F,
           # if TRUE run in parallel
           ncores = F,
           # if FALSE (default) use max number of cores - 1. Else set to number of
cores to use
           backwards = F) {
    # if FALSE (default) run forwards simulation from starting point. Else backwar
ds from end point

    require(tidyverse)
    require(terra)

    difference = 0

    if (backwards) {
      fdate = as.character(format(
        as.POSIXct(lubridate::ymd(fdate) - lubridate::hours(23), format = '%m/%d/%
Y %H:%M:%S'),
        format = '%Y%m%d'
      ))

      fhour = as.character(lubridate::hour(
        lubridate::hours(as.numeric(fhour)) - lubridate::hours(nforecast)
      ) %% 24)

      # define the interval
      interval <- c("00", "06", "18")
```

```
      # calculate the index of the closest interval
      index <-
        which.min(abs(as.numeric(fhour) - as.numeric(interval)))

      # calculate the difference
      difference <-
        as.numeric(fhour) - as.numeric(interval[index])

      # get the corresponding interval
      fhour <- interval[index]
    }

    pathway <- file.path(data_path, fdate, fhour)

    files <- gfs_names(path = pathway)

    if (parallel) {
      require(doSNOW)
      require(foreach)

      cores = parallel::detectCores(logical = T)
      if (!is.numeric(ncores))
        ncores <- cores[1] - 1
      else
        if (ncores > cores[1])
          stop("Number of cores must be equal to or less than available cores in s
ystem")

      ncores = min(ncores, length(coords))
      print(paste("Running in parallel using", ncores, "cores"))

      cl <- makeCluster(ncores)
      registerDoSNOW(cl)

      print("Simulating:")
      pb <-
        txtProgressBar(
          min = 0,
          max = length(coords),
          style = 3,
          char = "="
        )
      progress <- function(n)
        setTxtProgressBar(pb, n)
      opts <- list(progress = progress)

      npoint <-
        foreach(
          point = 1:length(coords),
          .packages = c("tidyverse", "terra"),
          .export = c(
            "read_u",
            "read_v",
            "wind_speed",
            "wind_direction",
            "next_cell"
```

```
    ),
    .options.snow = opts
  ) %dopar% {
    r <- terra::rast(file.path(data_path, files$file[1]))

    # extract coordinates
    long = coords[[point]][1]
    lat = coords[[point]][2]

    # empty raster for simulations
    fct_raster <- r
    fct_raster[] <- 0
    names(fct_raster) <- "wind_forecast"

    xlen <- terra::ncol(r)
    ylen <- terra::nrow(r)

    # weights for the output
    wt <- c(1, 1, 1, 1, 3, 1, 1, 1, 1)

    points_full <-
      tibble(
        x = numeric(),
        y = numeric(),
        nsim = numeric(),
        nforecast = numeric()
      )

    for (rep in seq_len(nsim)) {
      points <- data.frame(
        x = colFromX(r, long),
        y = rowFromY(r, lat),
        nforecast = 0
      )

      n <- 1

      for (f in seq_len(nforecast)) {
        x <- points[n, 1]
        y <- points[n, 2]

        forecasts <- unique(files$forecast)

        u <-
          read_u(
            path = data_path,
            files_list = files,
            fcast = forecasts[f],
            lev = atm_level
          )
        v <-
          read_v(
            path = data_path,
            files_list = files,
            fcast = forecasts[f],
            lev = atm_level
          )
```

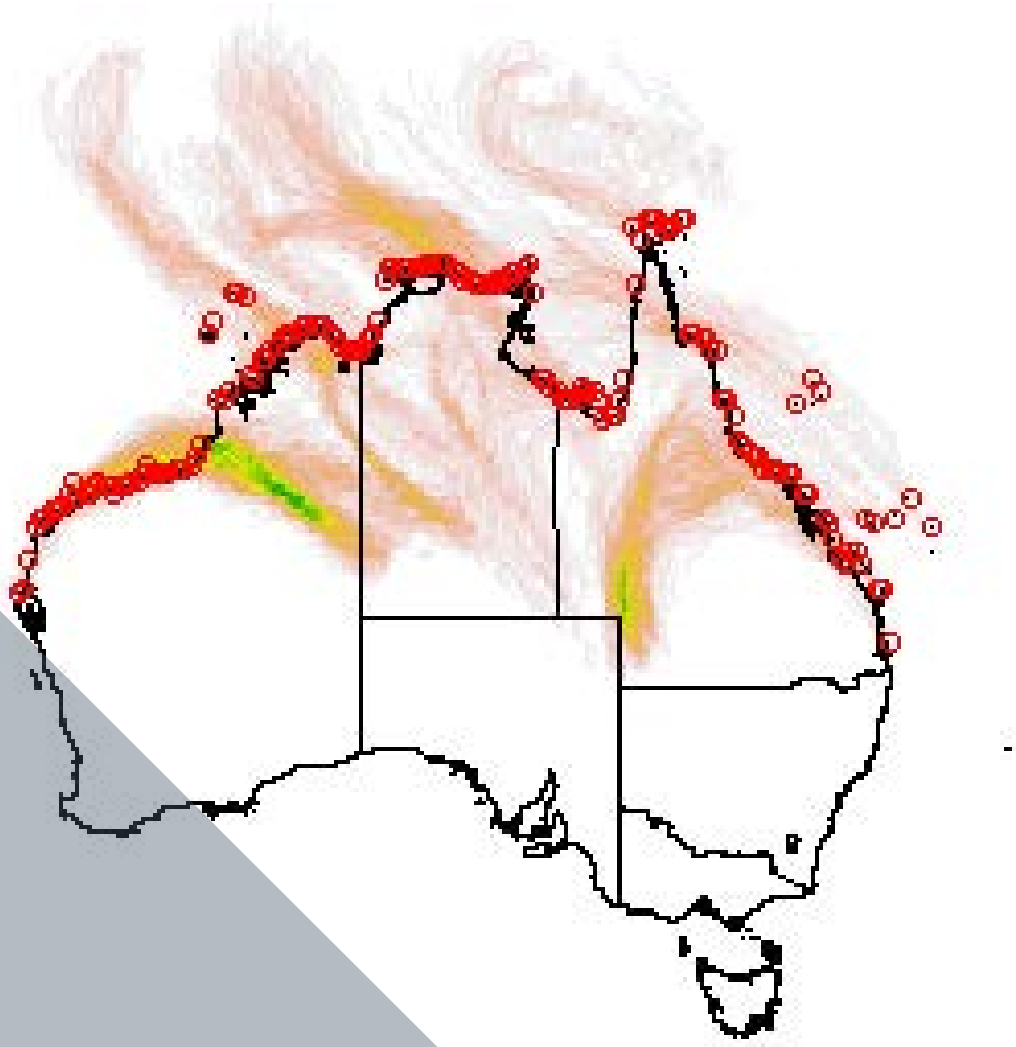
```
            # calculate wind speed and direction
            speed <- wind_speed(u = u, v = v)
            direction <- wind_direction(u = u, v = v)

            speed_ctr <- speed[y, x][1, 1]

            ## calculate the number of steps based on wind speed and cell size
            # if we choose at least 1 step each time there could be too many ste
ps overall
            # when the speed is low that results in overshooting, i.e. trajector
ies longer than reality
            # this could be happening because of coarse raster resolution
            # so I made it random, to have some movement with low wind speed, bu
t not always
            steps <-
              max(sample(0:1, 1), ceiling(speed_ctr * 3600 / cellsize))

            if (steps < 1)
              next

            for (e in seq_len(steps)) {
              nbr_dir <- c()
              nbr_spd <- c()
              for (i in c(-1, 0, 1)) {
                for (j in c(-1, 0, 1)) {
                  if (x + i < xlen && y + j < ylen) {
                    nbr_dir <- c(nbr_dir, direction[y + j, x + i][1, 1])
                    nbr_spd <-
                      c(nbr_spd, speed[y + j, x + i][1, 1])
                  }
                }
              }
              # multiply the weight with the speeds
              probs <- wt * nbr_spd
              # add some randomness to the direction
              selected_dir <-
                sample(x = nbr_dir,
                       size = 1,
                       prob = probs)
              selected_dir <-
                selected_dir + runif(1, -30, 30)
              # keep the random direction within 0-360
              selected_dir <- selected_dir %% 360
              # calculate the next point
              newpoint <- next_cell(selected_dir, x, y)
              fct_raster[newpoint[2], newpoint[1]] <-
                fct_raster[newpoint[2], newpoint[1]][1, 1] + 1

              n <- n + 1
              points[n, "x"] <- newpoint[1]
              points[n, "y"] <- newpoint[2]
              points[n, "nforecast"] <- f
            }
          }

          if (full)
```

```
            points_full <- bind_rows(
              points_full,
              as_tibble(xyFromCell(
                r,
                cellFromRowCol(r,
                               points[, "y"],
                               points[, "x"])
              )) %>%
                mutate(
                  nsim = rep,
                  nforecast = points$nforecast,
                  x_start = long,
                  y_start = lat
                )
            )
        }

        if (full)
          list(raster::raster(fct_raster), points_full)
        else
          raster::raster(fct_raster)
      }
    close(pb)
    stopCluster(cl)
  } else {
    r <- terra::rast(file.path(pathway, files$file[1]))

    npoint <- list()

    print("Simulating:")
    progress_bar = txtProgressBar(
      min = 0,
      max = length(coords) * nsim * nforecast,
      style = 3,
      char = "="
    )

    for (point in 1:length(coords)) {
      # extract coordinates
      long = coords[[point]][1]
      lat = coords[[point]][2]

      # empty raster for simulations
      fct_raster <- r
      fct_raster[] <- 0
      names(fct_raster) <- "wind_forecast"

      xlen <- terra::ncol(r)
      ylen <- terra::nrow(r)

      # weights for the output
      wt <- c(1, 1, 1, 1, 3, 1, 1, 1, 1)

      points_full <-
        tibble(
          x = numeric(),
          y = numeric(),
```

```
            nsim = numeric(),
            nforecast = numeric()
          )

        for (rep in seq_len(nsim)) {
          points <- data.frame(
            x = colFromX(r, long),
            y = rowFromY(r, lat),
            nforecast = 0
          )

          n <- 1

          forecast_hours <- seq_len(nforecast)
          if (backwards)
            forecast_hours <- rev(forecast_hours)

          for (f in forecast_hours) {
            x <- points[n, 1]
            y <- points[n, 2]

            forecasts <- unique(files$forecast)

            u <-
              read_u(
                path = pathway,
                files_list = files,
                fcast = forecasts[f + difference],
                lev = atm_level
              )
            v <-
              read_v(
                path = pathway,
                files_list = files,
                fcast = forecasts[f + difference],
                lev = atm_level
              )

            # calculate wind speed and direction
            speed <- wind_speed(u = u, v = v)
            direction <- wind_direction(u = u, v = v)

            if (backwards)
              direction <- (direction + 180) %% 360

            speed_ctr <- speed[y, x][1, 1]

            ## calculate the number of steps based on wind speed and cell size
            # if we choose at least 1 step each time there could be too many steps
overall
            # when the speed is low that results in overshooting, i.e. trajectorie
s longer than reality
            # this could be happening because of course raster resolution
            # so I made it random, to have some movement with low wind speed, but
not always
            steps <-
              max(sample(0:1, 1), ceiling(speed_ctr * 3600 / cellsize))
```

```
      if (steps < 1)
        next

      for (e in seq_len(steps)) {
        nbr_dir <- c()
        nbr_spd <- c()
        for (i in c(-1, 0, 1)) {
          for (j in c(-1, 0, 1)) {
            if (x + i < xlen && y + j < ylen) {
              nbr_dir <- c(nbr_dir, direction[y + j, x + i][1, 1])
              nbr_spd <-
                c(nbr_spd, speed[y + j, x + i][1, 1])
            }
          }
        }
        # multiply the weight with the speeds
        probs <- wt * nbr_spd
        # add some randomness to the direction
        selected_dir <-
          sample(x = nbr_dir,
                 size = 1,
                 prob = probs)
        selected_dir <- selected_dir + runif(1, -30, 30)
        # keep the random direction within 0-360
        selected_dir <- selected_dir %% 360
        # calculate the next point
        newpoint <- next_cell(selected_dir, x, y)
        fct_raster[newpoint[2], newpoint[1]] <-
          fct_raster[newpoint[2], newpoint[1]][1, 1] + 1

        n <- n + 1
        points[n, "x"] <- newpoint[1]
        points[n, "y"] <- newpoint[2]
        points[n, "nforecast"] <- f
      }

      setTxtProgressBar(progress_bar,
                        value = (point - 1) * nsim * nforecast +
                          (rep - 1) * nforecast +
                          which(forecast_hours == f))
    }

    if (full) {
      points_full <- bind_rows(
        points_full,
        as_tibble(xyFromCell(
          r,
          cellFromRowCol(r,
                         points[, "y"],
                         points[, "x"])
        )) %>%
          mutate(
            nsim = rep,
            nforecast = points$nforecast,
            x_start = long,
            y_start = lat
```

```
              )
          )
          npoint[[point]] <-
            list(raster::raster(fct_raster), points_full)

        } else
          npoint[[point]] <- raster::raster(fct_raster)
      }
    }
  }

  if (full) {
    fct_raster <- stack(lapply(npoint, "[[", 1))
    fct_raster <- raster::calc(fct_raster, sum)
    fct_raster[fct_raster == 0] <- NA
    points_full <- lapply(npoint, "[[", 2)
    return(list(rast(fct_raster), bind_rows(points_full)))
  } else {
    if (length(npoint) > 1) {
      fct_raster <-
        ifelse(length(npoint) > 1,
               stack(npoint),
               as(fct_raster, "Raster"))
      fct_raster <- raster::calc(fct_raster, sum)
    } else
      fct_raster <- raster::raster(fct_raster)

    fct_raster[fct_raster == 0] <- NA
    return(rast(fct_raster))
  }
}
```



Plant Health Australia
ABN 97 092 607 997
Level 1, 1 Phipps Close
Deakin ACT 2600

Phone 02 6215 7700
Email biosecurity@phau.com.au
planthealthaustralia.com.au

Plant Health AUSTRALIA

IF YOU SEE ANYTHING UNUSUAL, CALL THE EXOTIC PLANT PEST HOTLINE

1800 084 881

PHA18-075