

Chasing the wind:

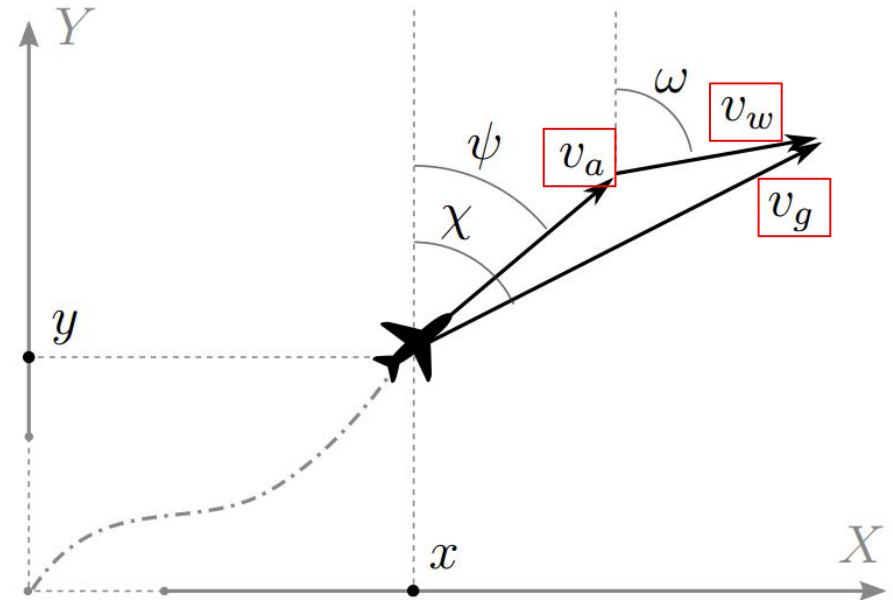
Ground-based Wind Field Construction from Mode-S and ADS-B Data with a Novel Gas Particle Model

Junzi Sun, Huy Vu, Joost Ellerbroek, Jacco Hoekstra

Control and Simulation, Faculty of Aerospace Engineering
Delft University of Technology, the Netherlands

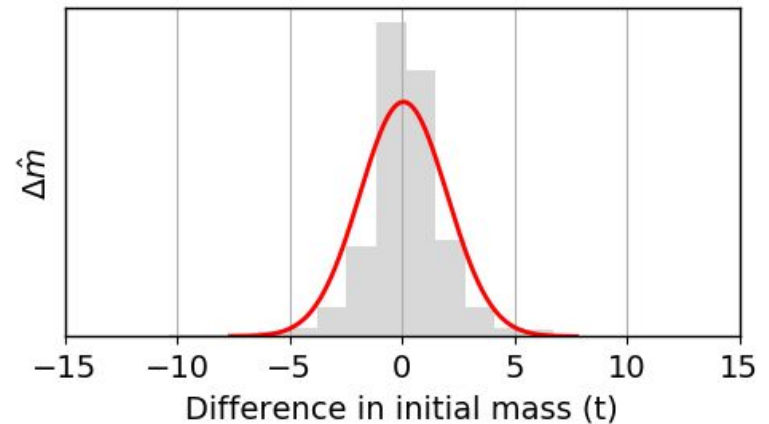
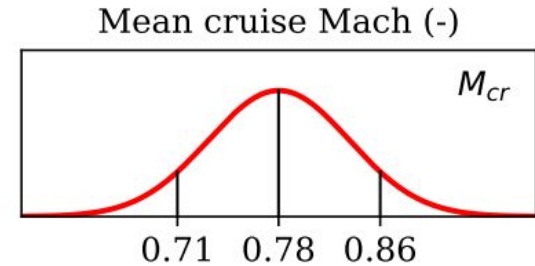
Why does wind matter for ATM?

- Trajectory prediction
- Optimize route for fuel saving
- ATM performance analysis
- Parameter estimations
- Aircraft performance modeling



Example when wind is not there:

1. Modeling cruise Mach number from ADS-B data.
2. Aircraft mass estimated based on ADS-B data with and without wind.



Why it is so difficult to get wind (correctly)?

1. The stochastic nature of wind

- Appearing **random** in large time and space scale
- But **predictable** in local area and short time range

2. Lack of observations

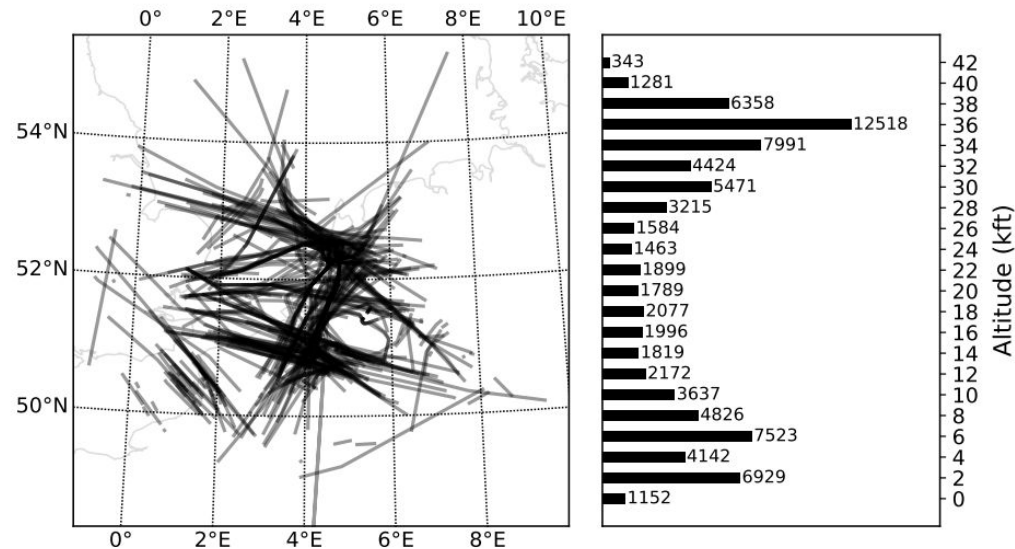
- Ground weather stations (accurate measurement, surface only, low update)
- Weather balloon (accurate measurement, fixed altitude, very small samples, low update)
- Remote sensing satellites (global coverage, less accurate, model dependent)
- Numerical weather prediction (NWP) models (combination of data sources, very low update rate, locally inaccurate due to smoothing and interpolations)

Which technologies are available for/in ATM?

1. Numerical weather prediction (NWP) models, used in many ATM researches
2. METAR for airports (from weather stations, typically update every 30 minute / 1 hour)
3. AMDAR (Aircraft Meteorological Data Relay)
4. MRAR (Mode-S Meteorological Routine Air Report aircraft observations)
5. Existing research categories based on aircraft:
 - a. Estimate wind from ground-based trajectories
 - b. Estimate local wind from aircraft Mode-S data
 - c. **Wind field estimation based on multiple wind measurements**

Challenges of using aircraft as sensors

1. Decoding of data in Mode-S
 - a. Detection of Mode-S message types
 - b. MRAR is very rarely available
 - c. Noise in aircraft data
2. Moving aircraft
 - a. Non-stationary measurements
3. Fixed flight route
 - a. Measurements are non-uniformly distributed in space and time



Distribution of 1-hour wind measurements

How does the model work?

What do the results look like?

How accurate is the model?

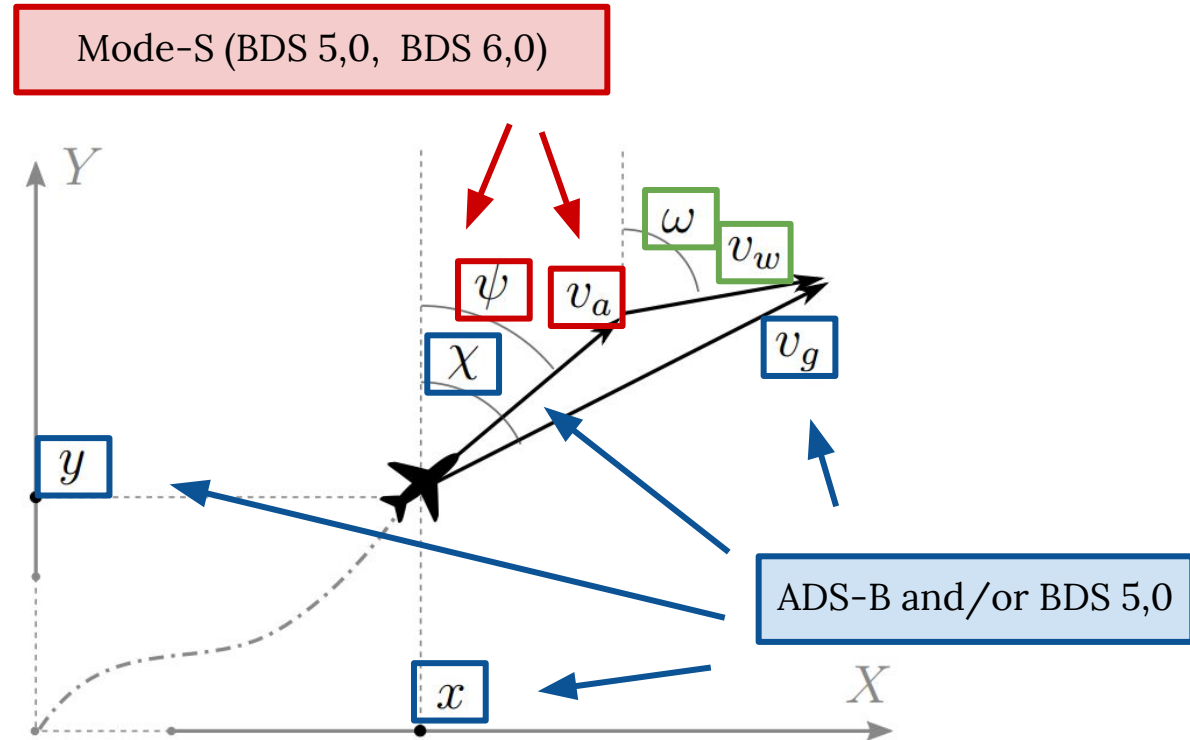
How does the model work?

Overview of the model

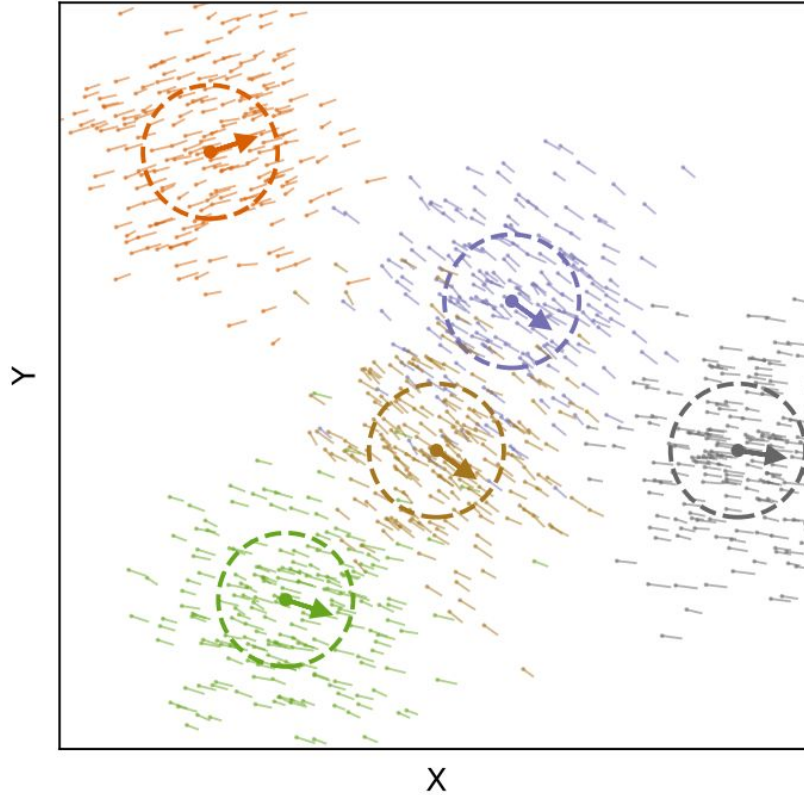
1. Getting the wind measurements
2. Generating particles
3. Particle motion model
4. Wind field reconstruction
5. Wind field confidence computation
6. Deletion of particles

Step 1: Getting the wind measurements

pyModeS



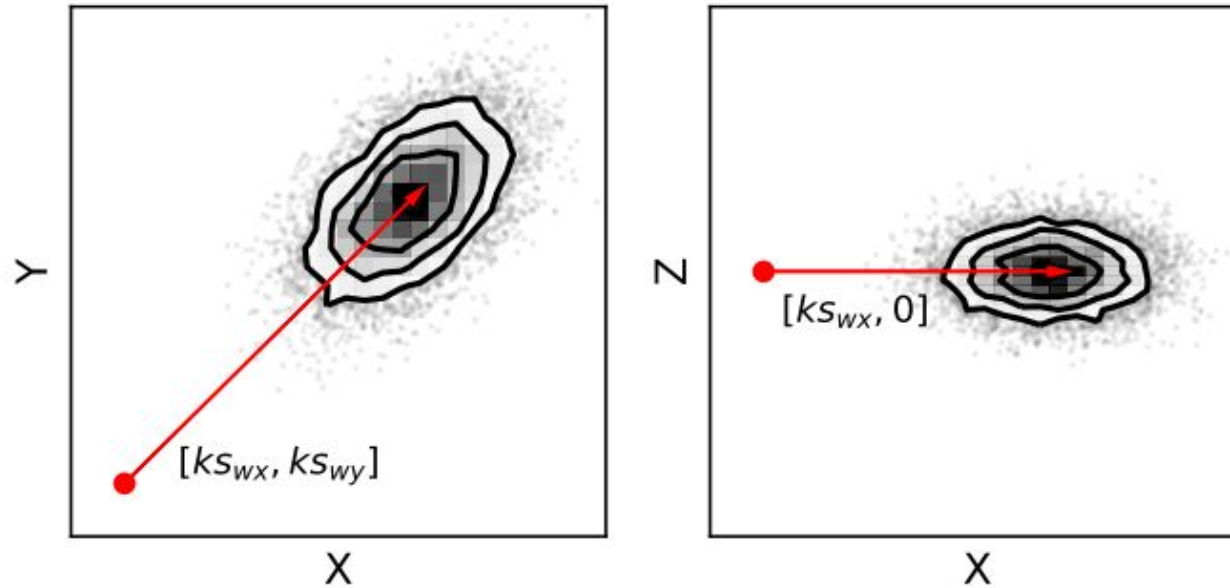
Step 2: Generating particles



$$\begin{pmatrix} p_{x,mn} \\ p_{y,mn} \\ p_{z,mn} \end{pmatrix} \sim \mathcal{N} \left(\begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix}, \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} \right)$$

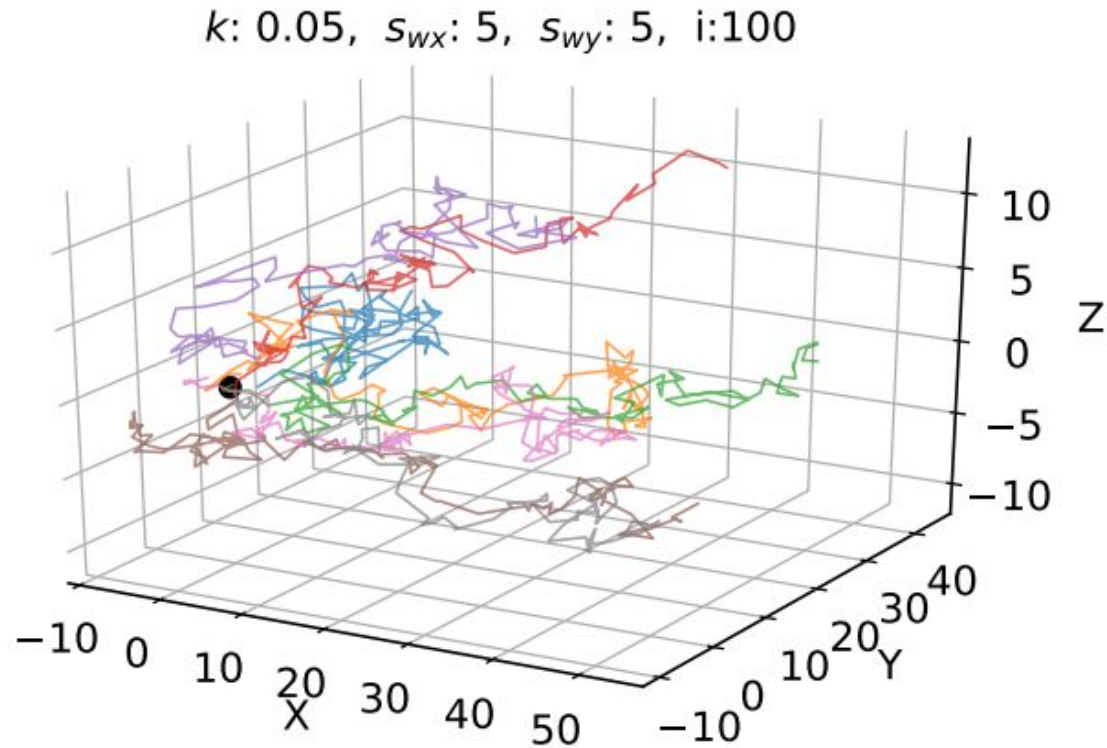
$$\begin{pmatrix} s_{wx,mn} \\ s_{wy,mn} \end{pmatrix} \sim \mathcal{N} \left(\begin{bmatrix} w_{x,m} \\ w_{y,m} \end{bmatrix}, \begin{bmatrix} \sigma_{wx}^2 & 0 \\ 0 & \sigma_{wy}^2 \end{bmatrix} \right)$$

Step 3: Particle motion model

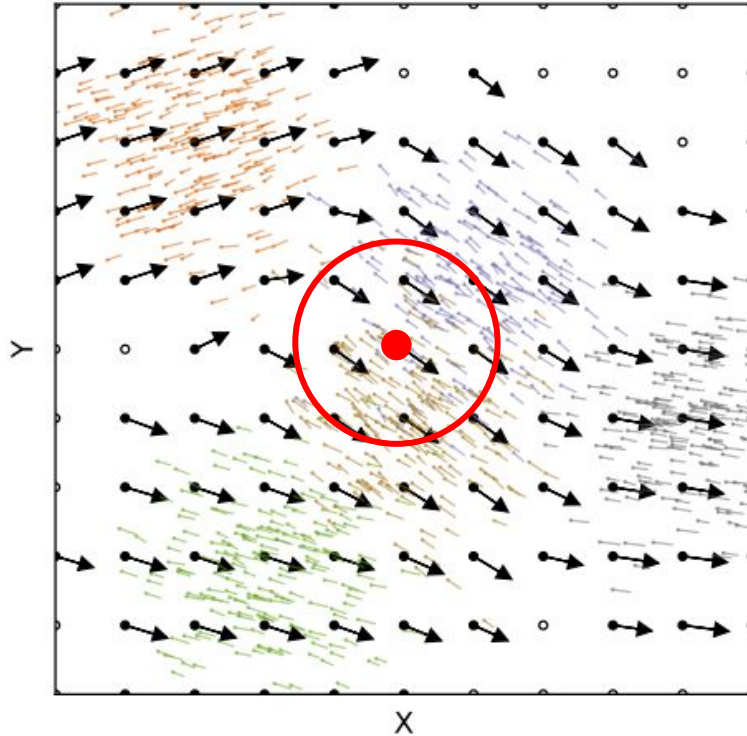


$$p(x,y,z)_{t+1} = p(x,y,z)_t + \epsilon_{(x,y,z)}$$
$$\epsilon_{(x,y)} \sim \mathcal{N} \left(\begin{bmatrix} k \cdot s_{wx} \\ k \cdot s_{wy} \end{bmatrix}, \begin{bmatrix} \sigma_{px}^2 & 0 \\ 0 & \sigma_{py}^2 \end{bmatrix} \right),$$
$$\epsilon_z \sim \mathcal{N}(0, \sigma_z^2)$$

Step 3: Particle motion model



Step 4: Wind field reconstruction

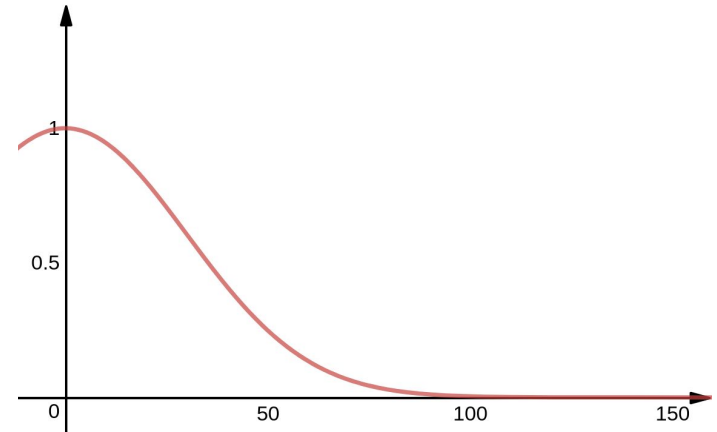


$$\begin{bmatrix} W_{x,(i,j,k)} \\ W_{y,(i,j,k)} \end{bmatrix} = \frac{1}{\sum_{p=1}^P \omega_p} \times \sum_{p=1}^P \left(\omega_p * \begin{bmatrix} s_{wx,p} \\ s_{wy,p} \end{bmatrix} \right)$$

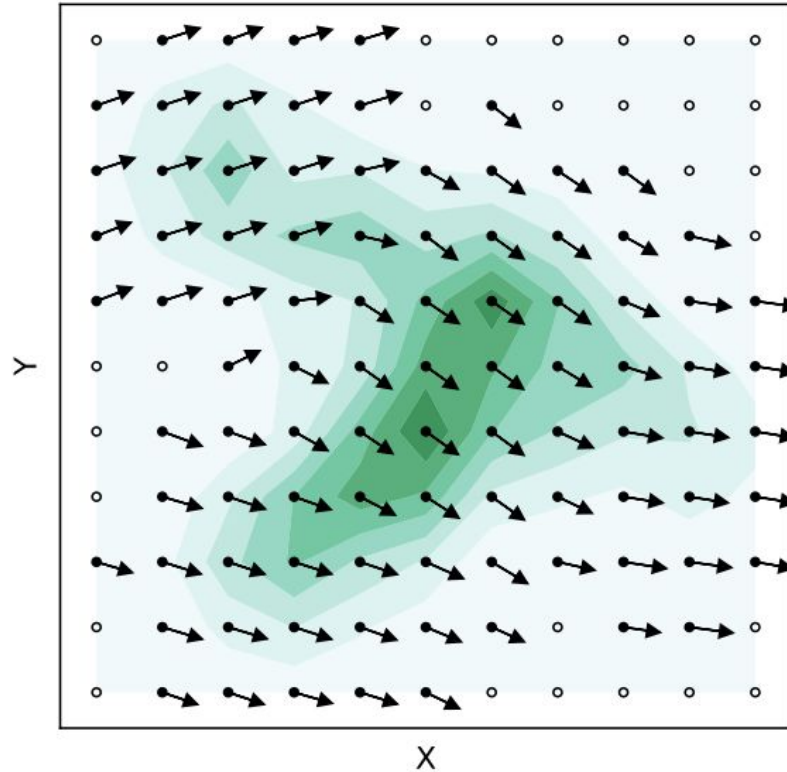
$$\omega_p = f_d(d) \times f_o(d_0) \times f_a(\tau)$$

Weights based on:

1. Distances to the positions of interest
2. Distances to the origin of gas particles
3. The age of the particles



Step 5: Wind field confidence model

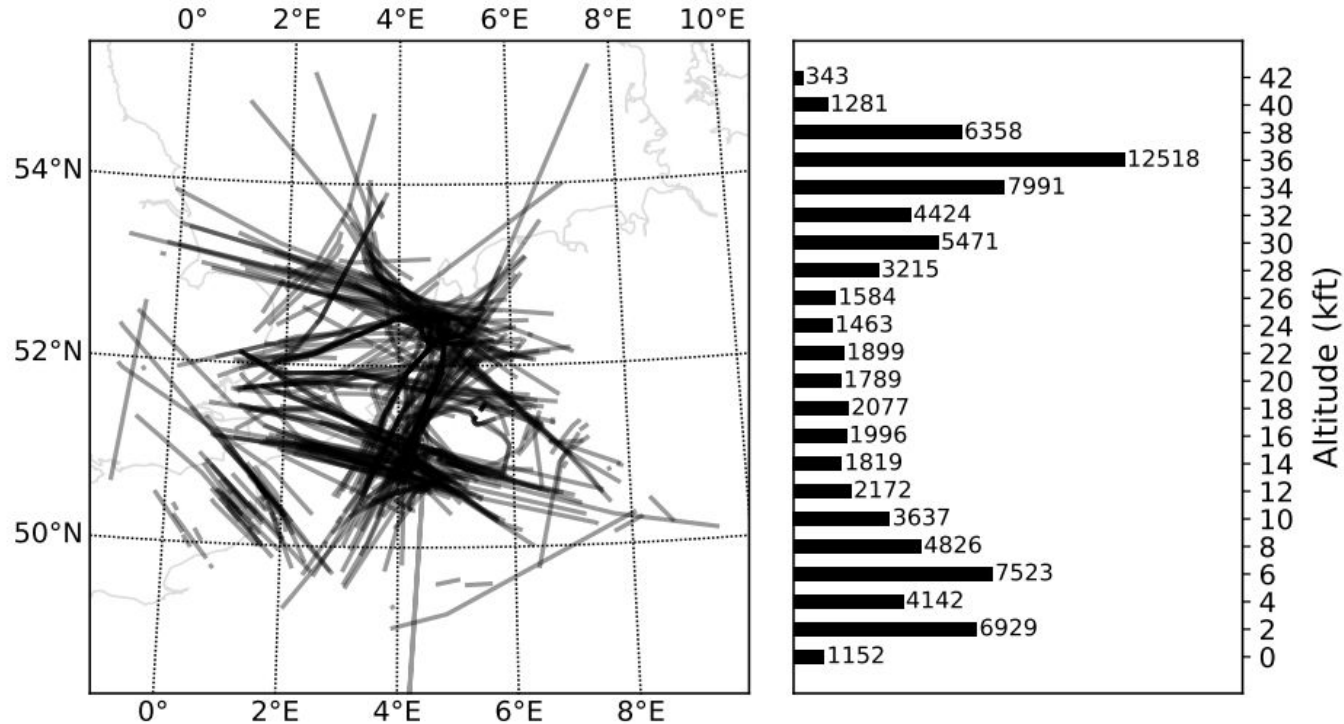


Combination of four factors:

1. the number of particles in the vicinity of the grid point (N)
2. the mean distances between particles and the grid point (D)
3. the homogeneity of wind states carried by particles (H)
4. the strength of particles due to decaying function (S)

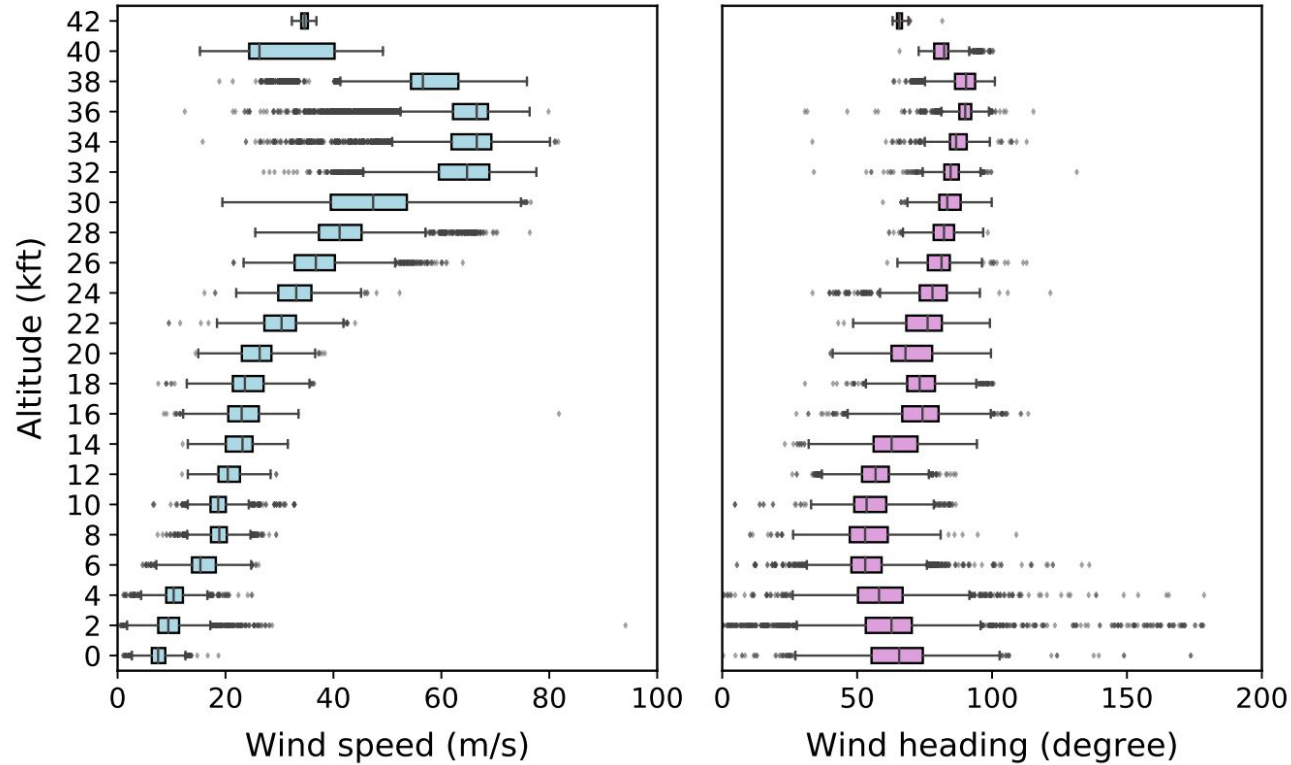
What do the results look like?

Experiment



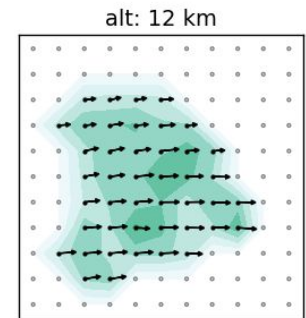
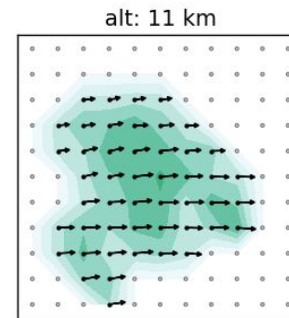
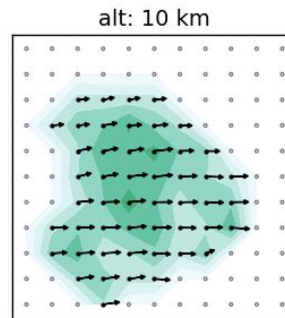
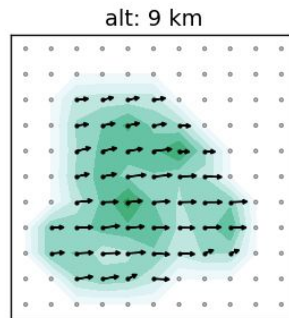
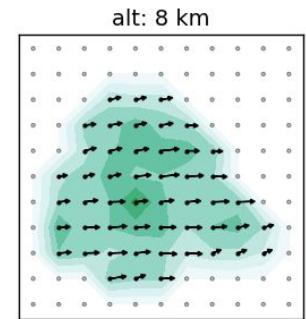
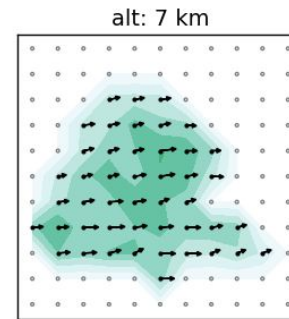
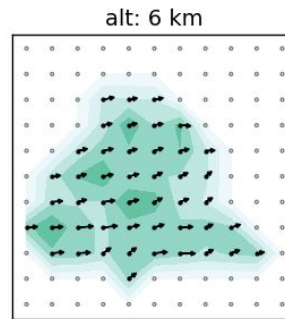
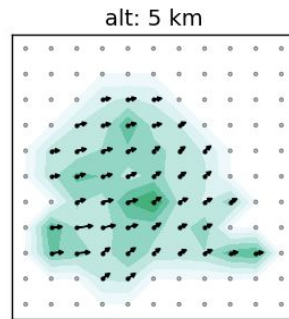
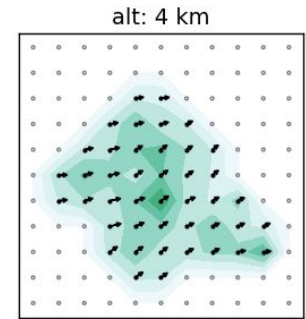
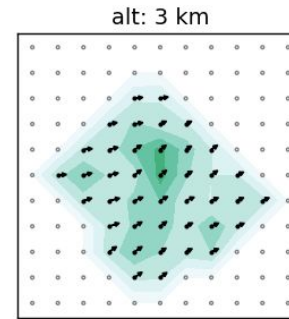
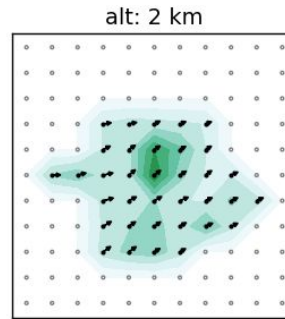
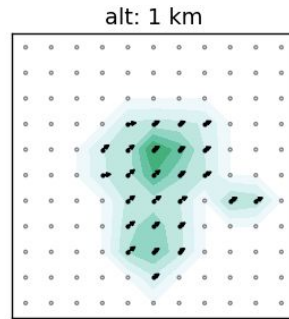
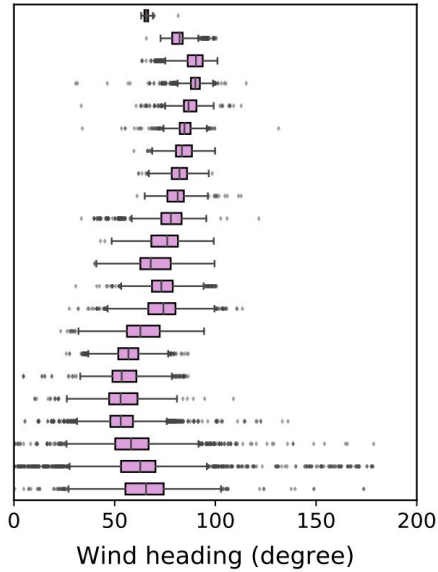
Distribution of 1-hour wind measurements (27/06/2017 12:00 UTC)

Experiment



Wind speed and heading distribution

Experiment



How accurate is the model?

Validation 1: Correctness

GFS Analysis dataset

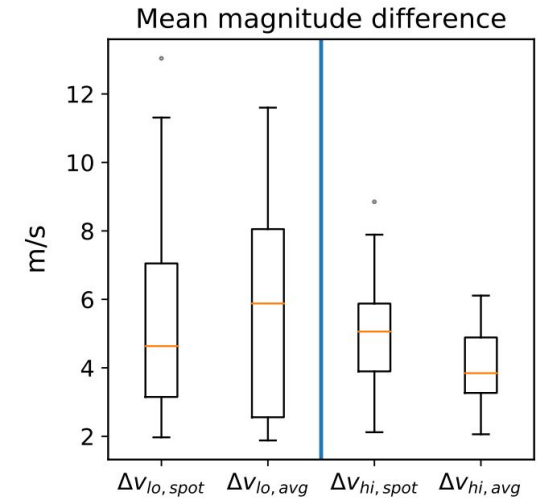
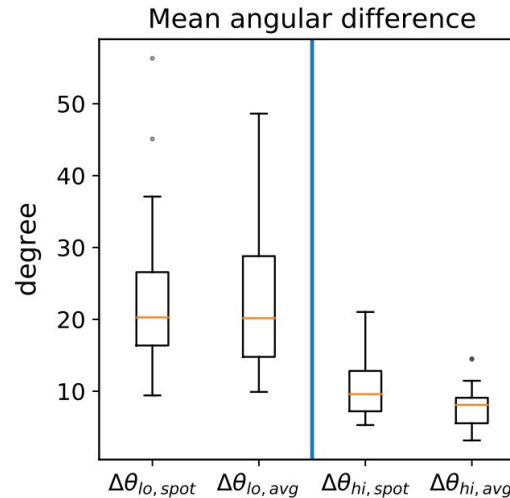
- 7 days (24 - 30 / 07 / 2017)
- 00:00, 06:00, 12:00, and 18:00 hour
- 0.25 degree resolution
- 19 different altitudes up to (~39 kft)

Comparisons

- Spot wind field
- Averaged wind field

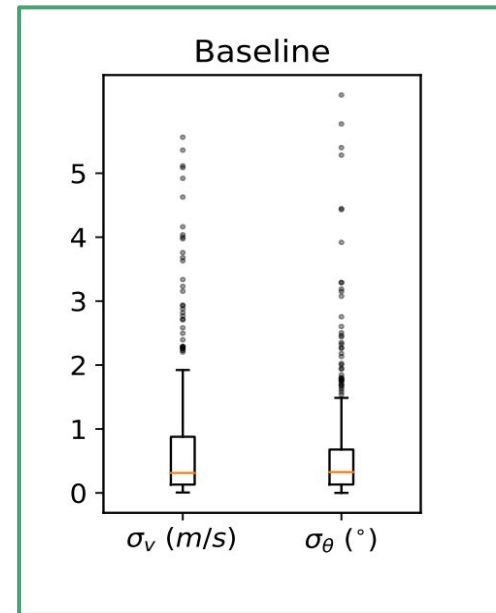
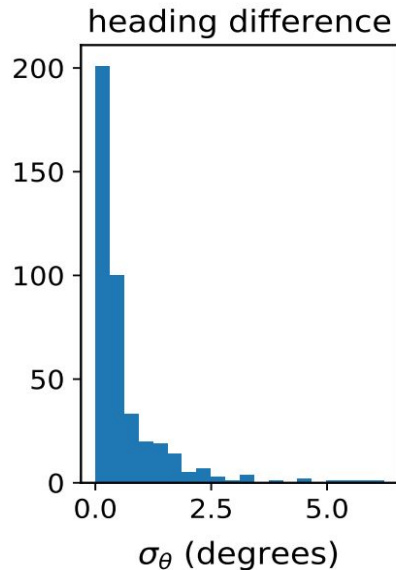
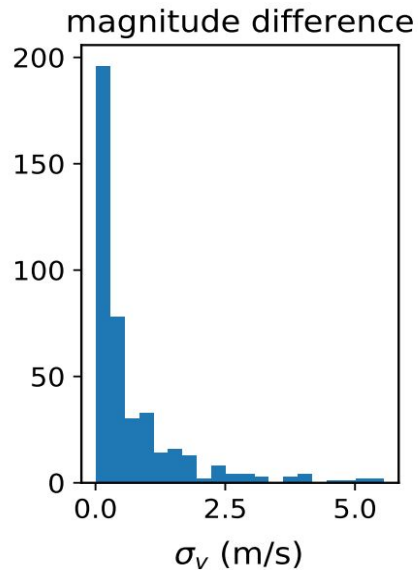
Groups

- Low wind group (< 10 m/s)
- High wind group (> 10 m/s)



Validation 2: Model variation

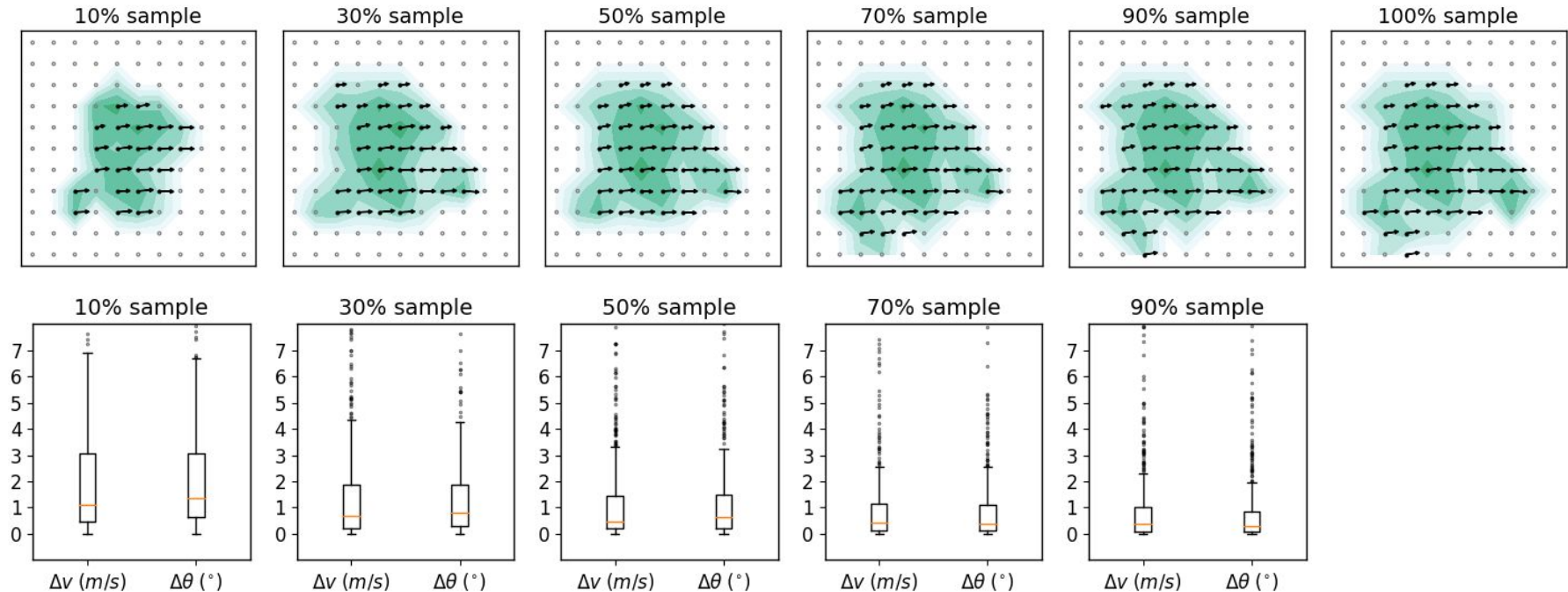
1. Capture the model variation due to stochastic process of the particles.
2. Multiple runs of the model on the same data



Standard deviation of wind speed and heading of 100 runs
(based on ~ 400 grid points)

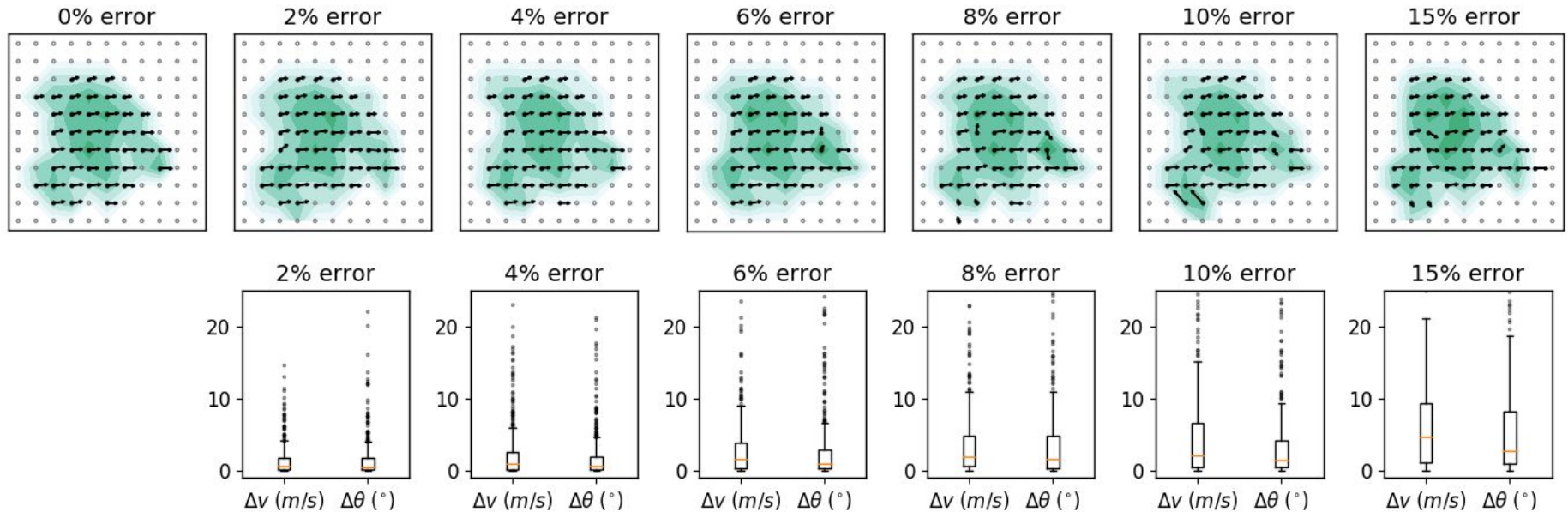
Validation 3: Data variation

1. Study how number of data samples effects the wind field
2. Executing model with random smaples of different percentage of measurement data



Validation 4: Robust error resistance

1. Study how error in data would affect the resulting wind field
2. Executing model with different percentage of measurement data **altered randomly**.



Conclusions

1. Developed a novel gas particle model for wind field construction
 - a. Particles to carry information
 - b. Wind field no-longer grid based
 - c. Fairly fast computation
 - d. For online processing, eliminate the need to store history wind measurements
2. Experiments and validations
 - a. Cross checked with GFS data
 - b. Self-validation of model variation, data variation, and robust error resistance
3. Improvement on pyModeS (message identification and decoding accuracy)
4. Open source library
 - a. <https://github.com/junzis/particle-wind-model>

Thanks for your attention!

Junzi Sun

Email: j.sun-1@tudelft.nl

GitHub: @junzis