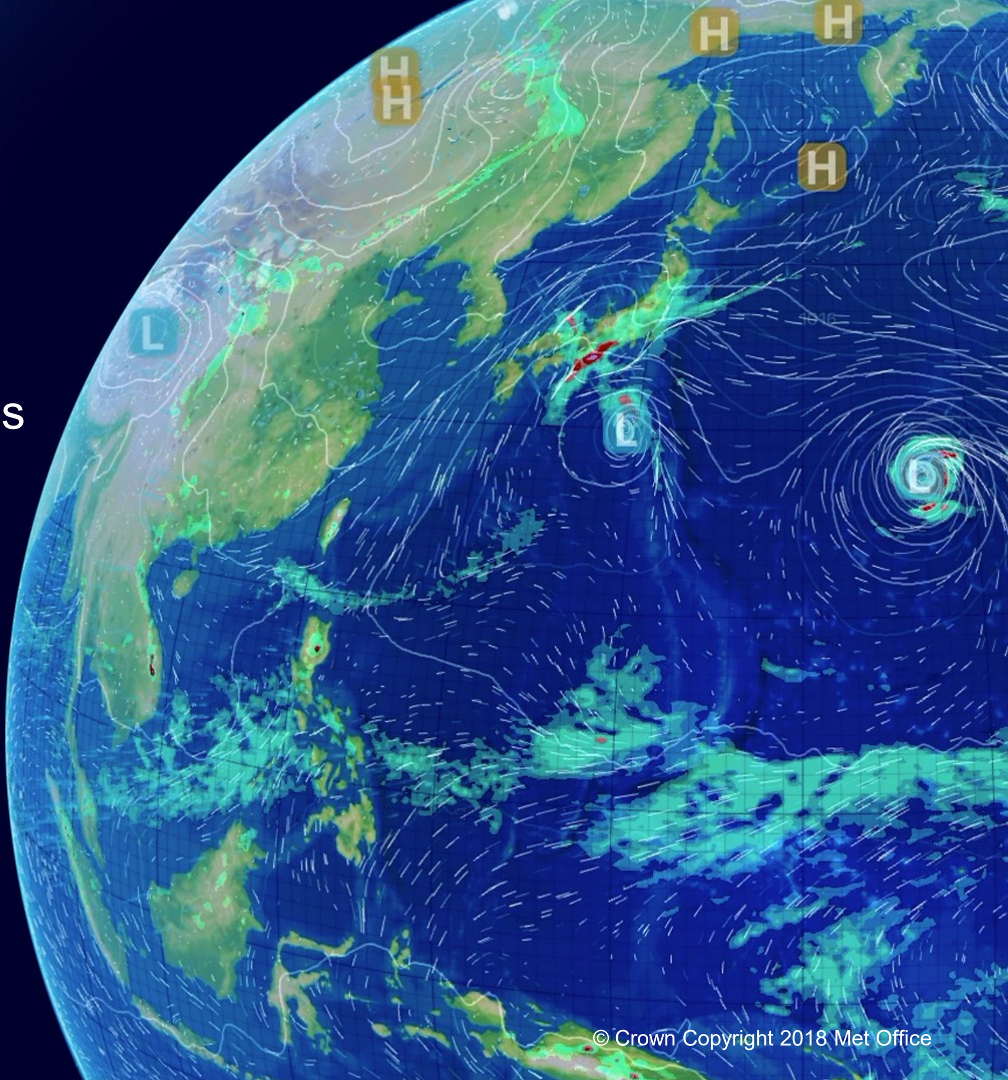
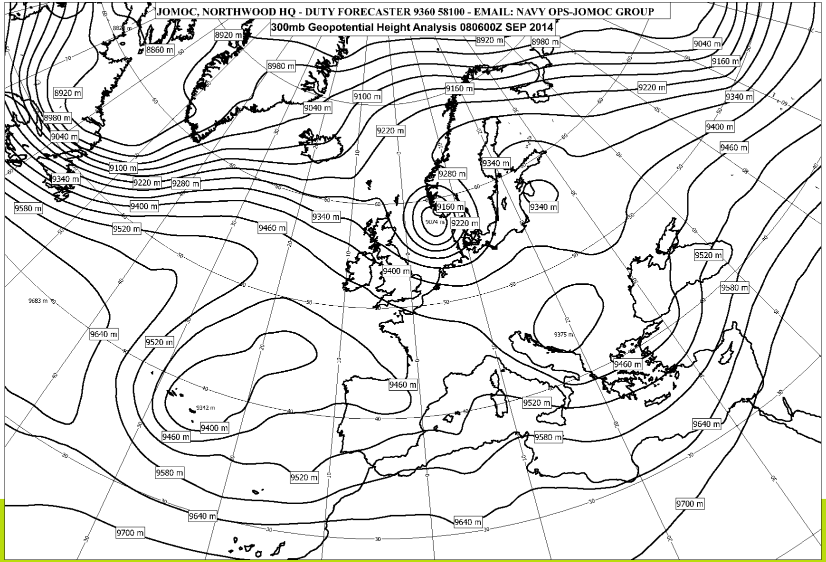
Machine Learning the Stability Function
in Land-Atmosphere Surface Exchanges

Cyril Morcrette, Martin Best,
Helena Reid, Joana Rodrigues, Theo Xirouchaki



Met Office

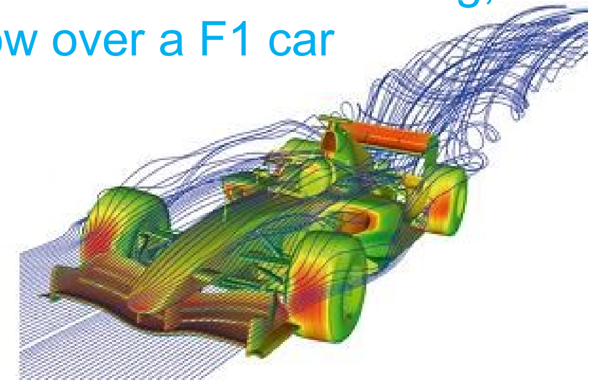
The atmosphere is a **fluid**

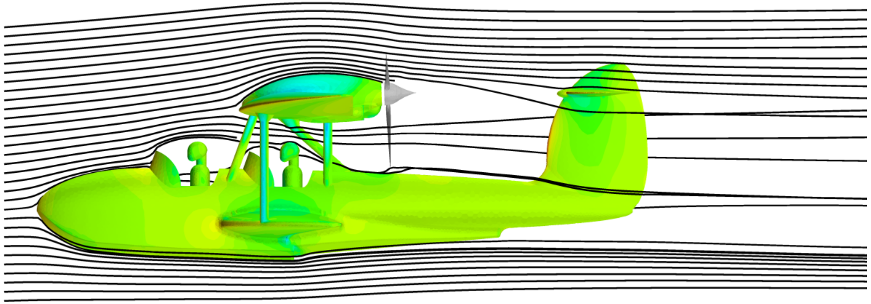


Solve equations of physics

1. Conservation of momentum:
   - Use Newton's Second law applied to fluid motion ("Navier–Stokes" equation).

Same equations as used for
flow through a pipe,
flow over an aircraft wing,
flow over a F1 car





Met Office

1. So take Navier–Stokes equation (conservation of momentum).

    - Put it on a sphere.
    - Allow the sphere to rotate.
    - Assume fluid depth << radius of the sphere (~20 km vs 6371 km)
    - Assume that acceleration of vertical wind << acceleration of horizontal components.

2. Continuity equation: represents the conservation of mass.

3. Conservation of energy: change in temperature of the system is related to the sources and sinks of heat.

These are the “primitive equations”

Met Office

The Primitive Equations

Equations of (horizontal) motion

$$\frac{\partial u}{\partial t}+\frac{u}{a\cos\phi}\frac{\partial u}{\partial\lambda}+\frac{v}{a}\frac{\partial u}{\partial\phi}+w\frac{\partial u}{\partial z}-\frac{uv}{a}\tan\phi=fv-\frac{1}{\rho a\cos\phi}\frac{\partial p}{\partial\lambda}+\mathscr{F}_1 \qquad \text{(a)}$$

$$\frac{\partial v}{\partial t}+\frac{u}{a\cos\phi}\frac{\partial v}{\partial\lambda}+\frac{v}{a}\frac{\partial v}{\partial\phi}+w\frac{\partial v}{\partial z}+\frac{u^2}{a}\tan\phi=-fu-\frac{1}{\rho a}\frac{\partial p}{\partial\phi}+\mathscr{F}_2 \qquad \text{(b)}$$

Hydrostatic Equilibrium equation

$$\frac{1}{\rho}\frac{\partial p}{\partial z}=-g=\frac{\partial\Phi}{\partial z} \qquad \text{(c)}$$

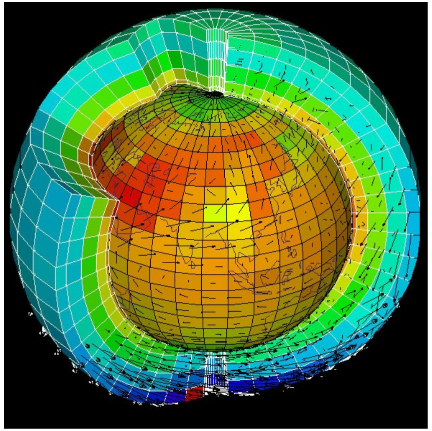
Continuity Equation

$$\frac{1}{a\cos\phi}\frac{\partial u}{\partial\lambda}+\frac{1}{a\cos\phi}\frac{\partial(v\cos\phi)}{\partial\phi}+\frac{w}{\rho(z)}\frac{\partial\left(\rho(z)w\right)}{\partial z}=0 \qquad \text{(d)}$$

Thermodynamic equation

$$\frac{\partial\theta}{\partial t}+\frac{u}{a\cos\phi}\frac{\partial\theta}{\partial\lambda}+\frac{v}{a}\frac{\partial\theta}{\partial\phi}+w\frac{\partial\theta}{\partial z}=l \qquad \text{(f)}$$

So define some **3-dimensional arrays** to represent the wind, temperature, pressure, water vapour, clouds etc at all latitudes and longitude and at a range of heights.



$$\frac{\partial\theta}{\partial t}+\frac{u}{a\cos\phi}\frac{\partial\theta}{\partial\lambda}+\frac{v}{a}\frac{\partial\theta}{\partial\phi}+w\frac{\partial\theta}{\partial z}=l$$

$$\frac{\partial\theta}{\partial t}=l-\frac{u}{a\cos\phi}\frac{\partial\theta}{\partial\lambda}-\frac{v}{a}\frac{\partial\theta}{\partial\phi}-w\frac{\partial\theta}{\partial z}$$

$$\frac{\theta(t+1)-\theta(t)}{\Delta t}=l-\frac{u}{a\cos\phi}\frac{\theta(i+1)-\theta(i-1)}{\Delta\lambda}-\frac{v}{a}\frac{\theta(j+1)-\theta(j-1)}{\Delta\phi}-w\frac{\theta(k+1)-\theta(k-1)}{\Delta z}$$

$$\theta(t+1)=\theta(t)+\Delta t\left\{l-\frac{u}{a\cos\phi}\frac{\theta(i+1)-\theta(i-1)}{\Delta\lambda}-\frac{v}{a}\frac{\theta(j+1)-\theta(j-1)}{\Delta\phi}-w\frac{\theta(k+1)-\theta(k-1)}{\Delta z}\right\}$$

Then solve the equations of motion *numerically* to see how everything evolves forward in time.

Then take the new atmospheric state at the new time and solve for the evolution over the next time-step. Then repeat lots of times.

How short does a time-step need to be? Typically ~10 minutes for global model.

So a 5 day forecast represents going forward 5*24*6=720 timesteps

Why does the time-step need to be so short?

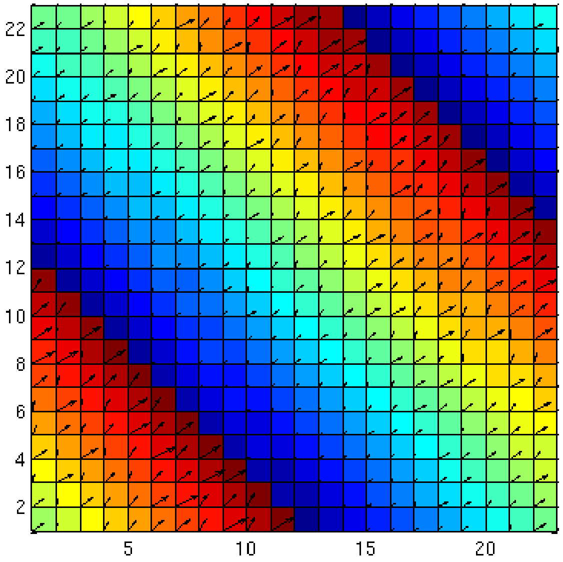
When solving an advection equation numerically, there is the Courant–Friedrichs–Lewy (CFL) criterion.

Basically, we must ensure the distance something moves over 1 time-step is no more than width of grid boxes.

Assuming a wind-speed of 20 m/s and a grid-box size of 20 km this means keeping the time-step to less than 1000s (i.e. 16 minutes).

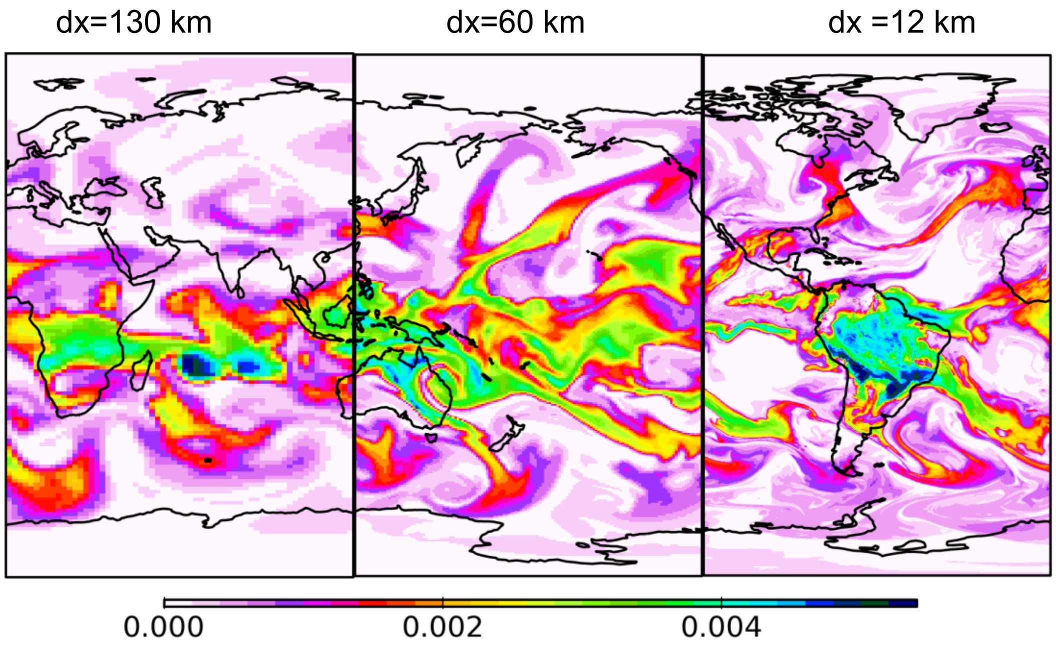
Otherwise, can get numerical artefacts such as:

- Negative concentrations!
- Fractional coverage (e.g. of cloudiness) which is <0 or >1
- Ripples near sharp gradients (not good for fronts)
- Lack of conservation! (total amount of mass, water etc not constant)

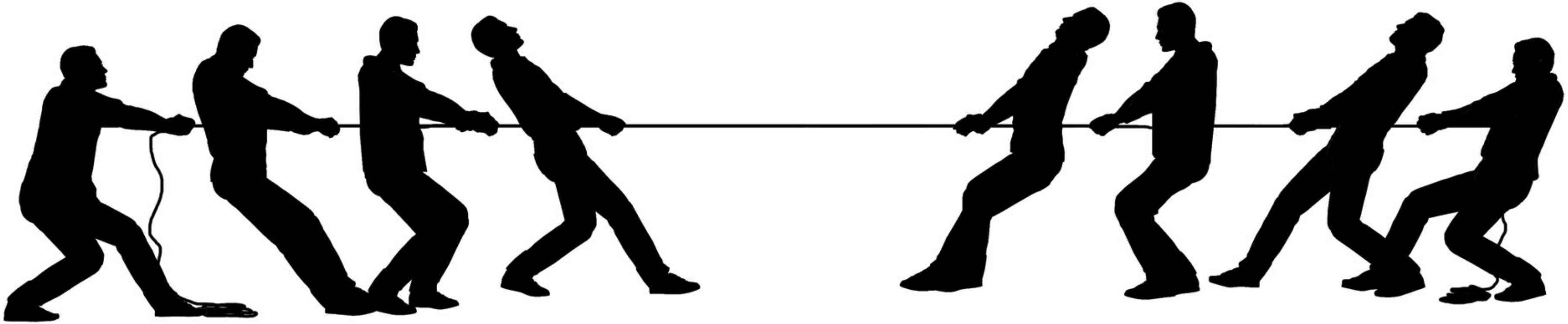


More resolution = More details

Humidity (kg/kg) at 500 hPa



So 2 conflicting aspirations



We want the grid as fine as possible in order to get as much detail as possible.

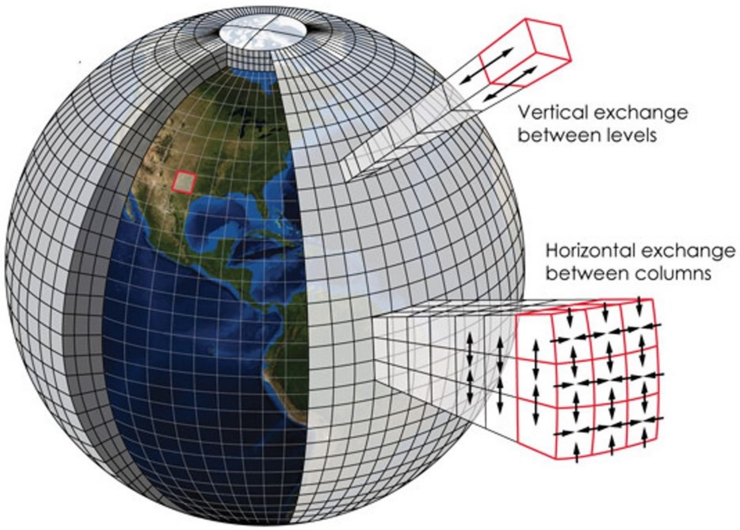
But atmosphere is 3 dimensional so <u>halving</u> the grid-length
(e.g. going from 40 km to 20 km and from 400m layers to 200m layers)
means 2 x 2 x 2 = 8 more data points,
so 8 times as much memory and 8 times as many calculations.

And (since the typical wind speed experienced on Earth does not change) halving the size of the grid-box also means having to halve the timestep.

So 8 times as many calculation 2 x as often so 16 times more expensive !

Met Office

Our global model is used for both WEATHER & CLIMATE



Global climate model has a horizontal resolution of 150 km

(This is run out for decades to look at future climate).

Global weather forecast model has a horizontal resolution of 15 km

(This is run out for 10 days).

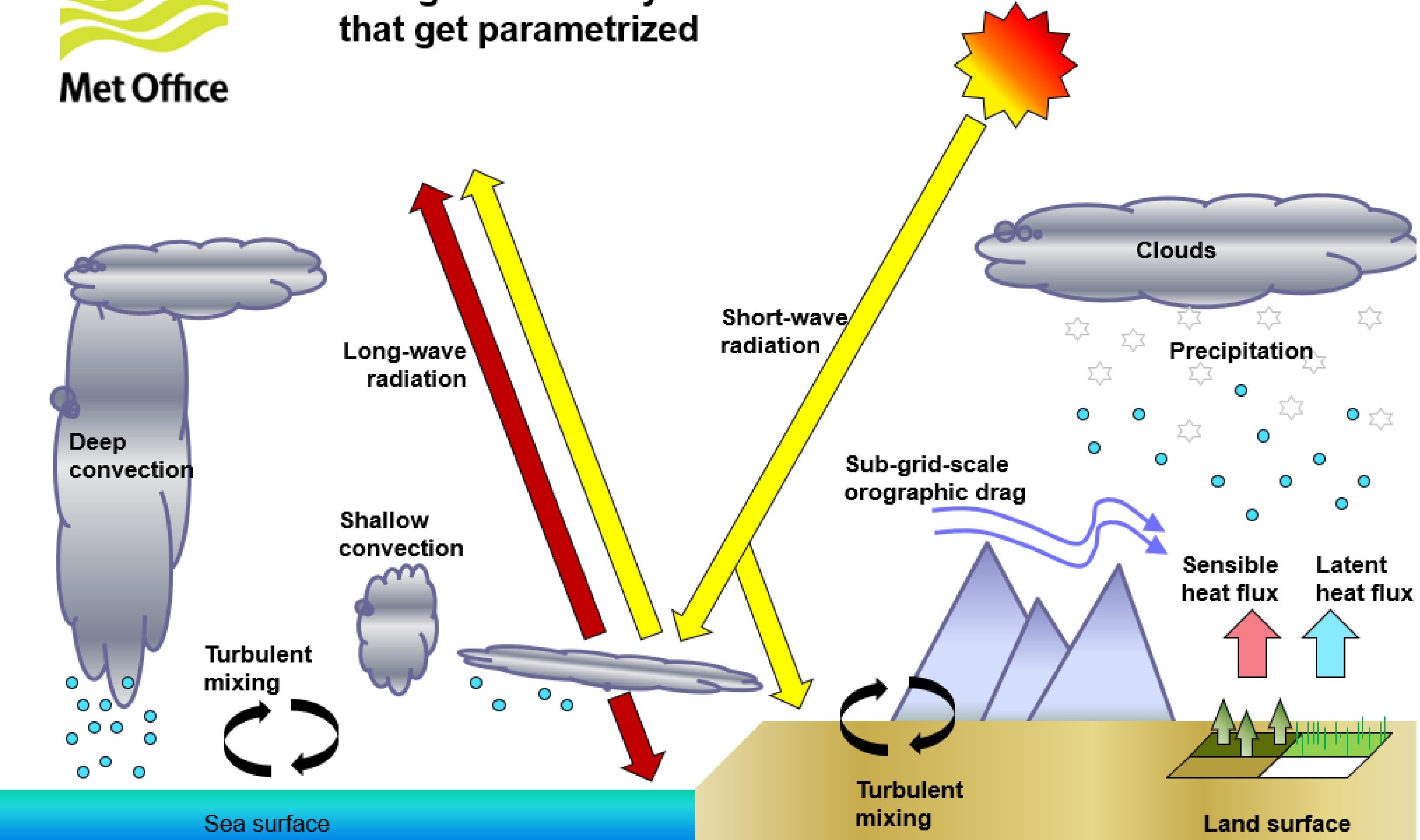
Both models have 70 levels in the vertical, going up to 80km.

Levels are thin near the surface and get progressively thicker with height.

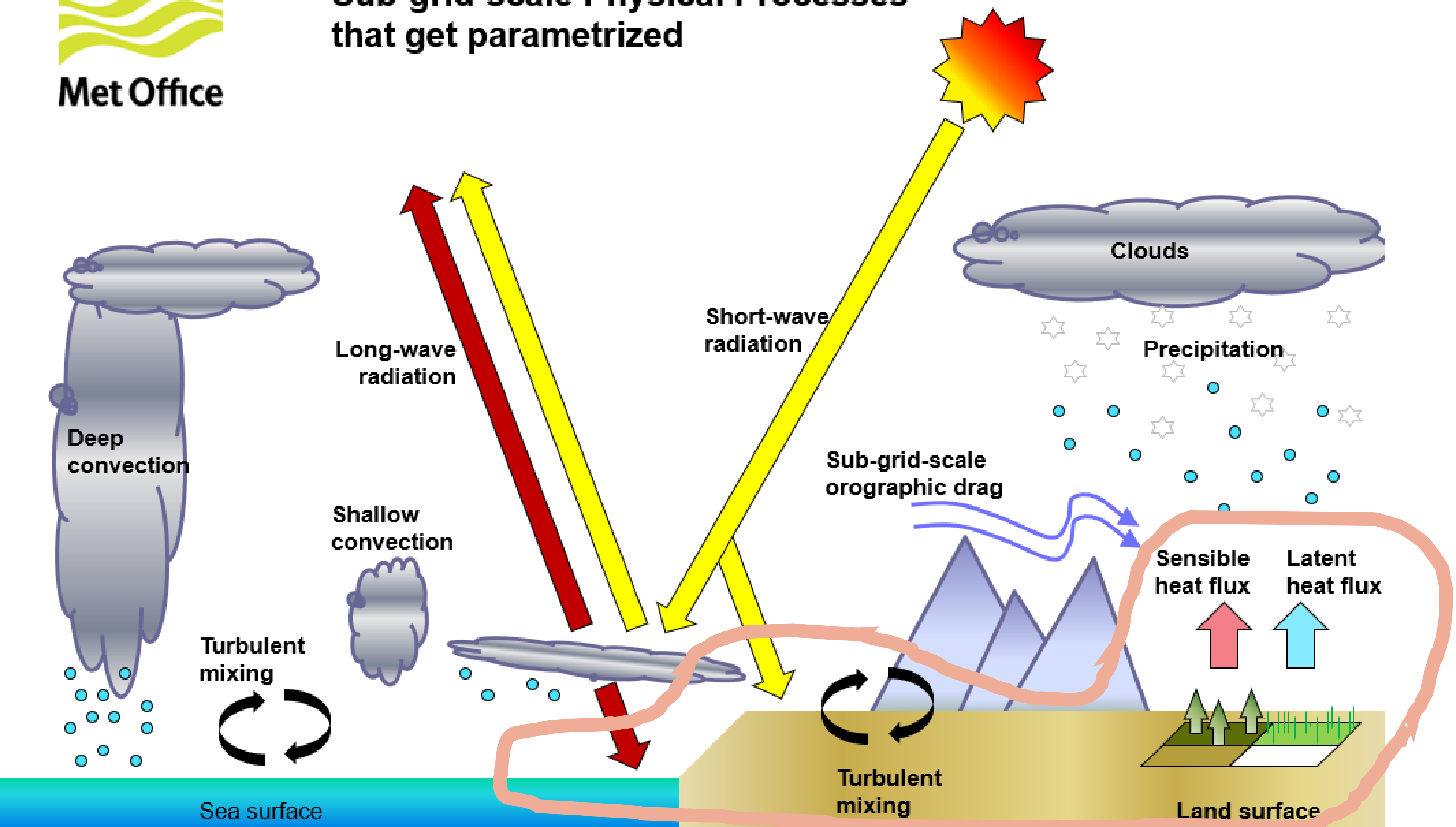
Lowest level is at 20m, 2nd lowest at 53m

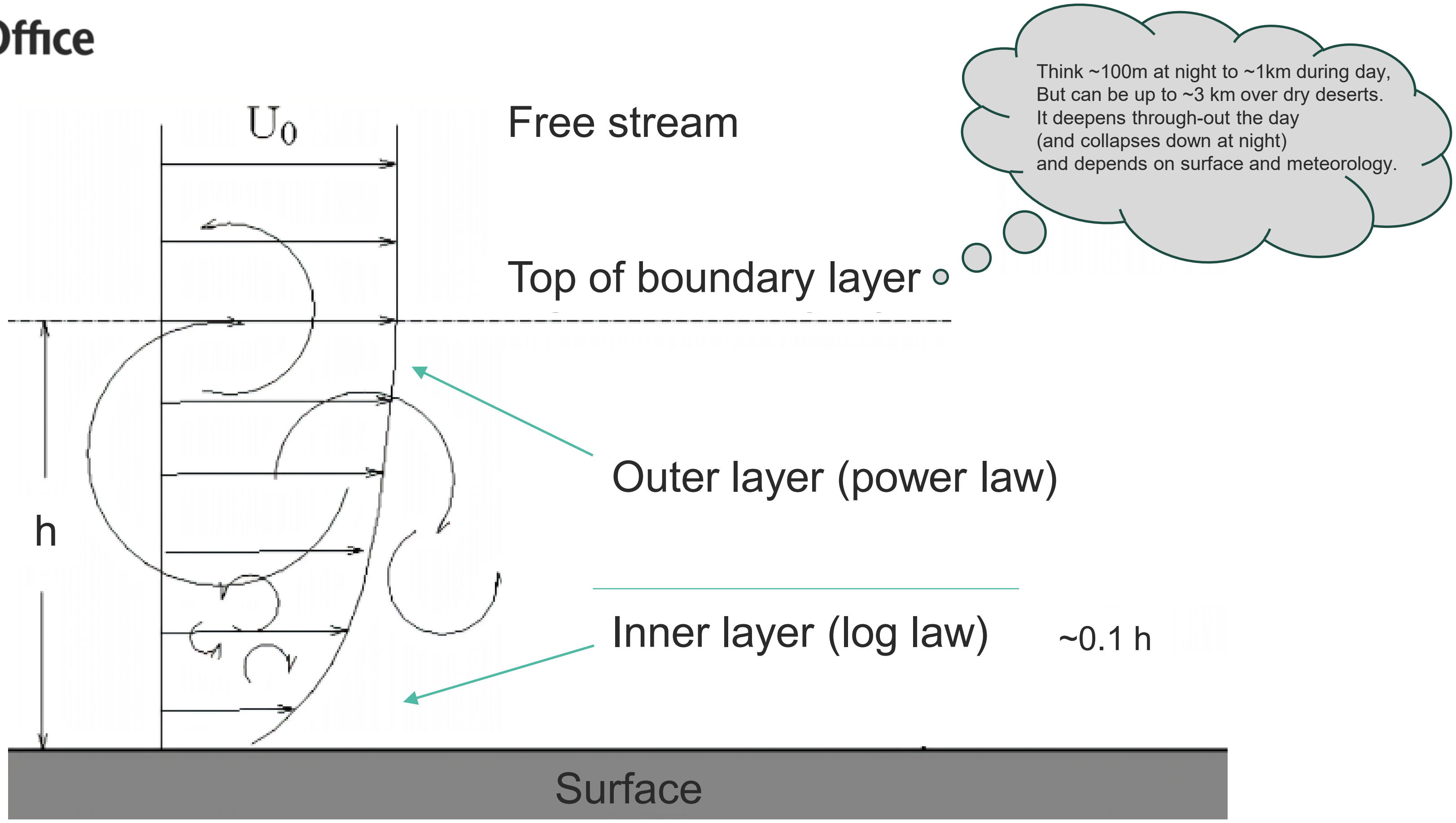
So can't resolve the details of say wind flowing over houses and trees.

Sub-grid-scale Physical Processes that get parametrized



Sub-grid-scale Physical Processes that get parametrized




Met Office
U_0
Free stream
Top of boundary layer
Think ~100m at night to ~1km during day,
But can be up to ~3 km over dry deserts.
It deepens through-out the day
(and collapses down at night)
and depends on surface and meteorology.
h
Outer layer (power law)
Inner layer (log law)
~0.1 h
Surface

Met Office

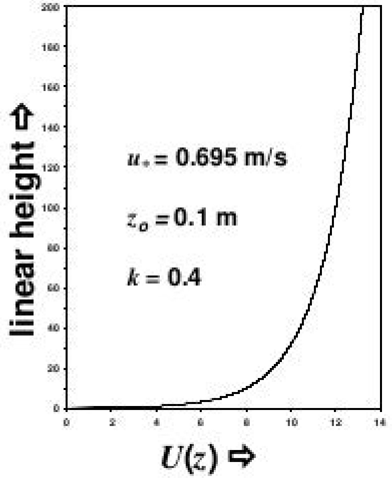
The Log Wind Profile

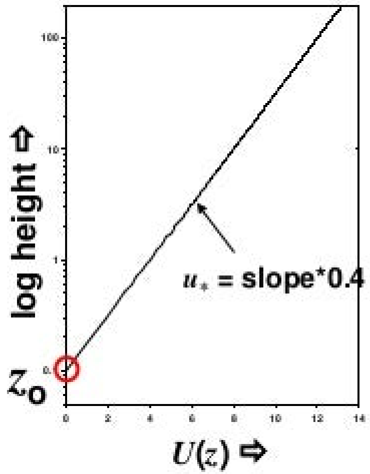
Under the assumptions of:

- no heat flux and
- statically neutral boundary layer (i.e. everything in steady balance)

The wind speed can be expressed as:

$$U(z) = \frac{u_*}{k} \ln\left[\frac{z}{z_o}\right]$$





Where z0 is the "roughness length" (notional height above surface where log wind profile goes to zero).

… but in general case, where there

IS a heat flux

Or

the flow is NOT neutral (so stable or unstable)

then

$$u_z = \frac{u_*}{\kappa}\left[\ln\left(\frac{z}{z_0}\right) + \psi(z, z_0, L)\right]$$

This is the “stability function”

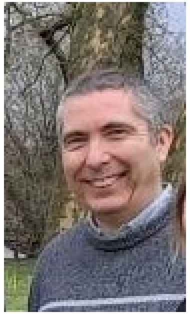
Met Office

… but in general case, where there
IS a heat flux
Or
the flow is NOT neutral (so stable or unstable)
then

$$u_z = \frac{u_*}{\kappa}\left[\ln\left(\frac{z}{z_0}\right) + \psi(z, z_0, L)\right]$$

This is the “stability function”

Our colleague, Martin Best, has gone through all the data
and using surface heat and momentum fluxes identified when the
conditions are neutral.
Hence ψ stability function=0
Hence can back out z0 from wind profile.



… but in general case, where there
IS a heat flux
Or
the flow is NOT neutral (so stable or unstable)
then

$$u_z = \frac{u_*}{\kappa}\left[\ln\left(\frac{z}{z_0}\right) + \psi(z, z_0, L)\right]$$

This is the "stability function"

Several functions have been proposed in the literature (e.g. Debolskiy et al 2023 for a review).

e.g. the "stability function" can be expressed in terms of the Richardson number, itself a function of the vertical gradients in temperature and wind speed.

This is what we are going to try to machine learn.

Something similar has been tried before.

Using ML and data from 2 sites.

Met Office

Extending the work of:

[Machine Learning for Improving Surface-Layer-Flux Estimates | Boundary-Layer Meteorology (springer.com)]

Boundary-Layer Meteorology (2022) 185:199–228
https://doi.org/10.1007/s10546-022-00727-4

RESEARCH ARTICLE

Check for updates

Machine Learning for Improving Surface-Layer-Flux Estimates

Tyler McCandless[1] · David John Gagne[2] · Branko Kosović[2] · Sue Ellen Haupt[2] · Bai Yang[3,4] · Charlie Becker[2] · John Schreck[2]

Received: 23 August 2021 / Accepted: 8 June 2022 / Published online: 13 September 2022

Abstract
Flows in the atmospheric boundary layer are turbulent, characterized by a large Reynolds number, the existence of a roughness sublayer and the absence of a well-defined viscous layer. Exchanges with the surface are therefore dominated by turbulent fluxes. In numerical models for atmospheric flows, turbulent fluxes must be specified at the surface; however, surface fluxes are not known a priori and therefore must be parametrized. Atmospheric flow models, including global circulation, limited area models, and large-eddy simulation, employ Monin–Obukhov similarity theory (MOST) to parametrize surface fluxes. The MOST approach is a semi-empirical formulation that accounts for atmospheric stability effects through universal stability functions. The stability functions are determined based on limited observations using simple regression as a function of the non-dimensional stability parameter representing a ratio of distance from the surface and the Obukhov length scale (Obukhov in Trudy Inst Theor Geofiz AN SSSR 1:95–115, 1946), z/L. However, simple regression cannot capture the relationship between governing parameters and surface-layer structure under the wide range of conditions to which MOST is commonly applied. We therefore develop, train, and test two machine-learning models, an artificial neural network (ANN) and random forest (RF), to estimate surface fluxes of momentum, sensible heat, and moisture based on surface and near-surface observations. To train and test these machine-learning algorithms, we use several years of observations from the Cabauw mast in the Netherlands and from the National Oceanic and Atmospheric Administration's Field Research Division tower in Idaho. The RF and ANN models outperform MOST. Even when we train the RF and ANN on one set of data and apply them to the second set, they provide more accurate estimates of all of the fluxes compared to MOST. Estimates of sensible heat and moisture fluxes are significantly improved, and model interpretability techniques highlight the logical physical relationships we expect in surface-layer processes.

We want to use a lot more data, from many more sites.

Met Office

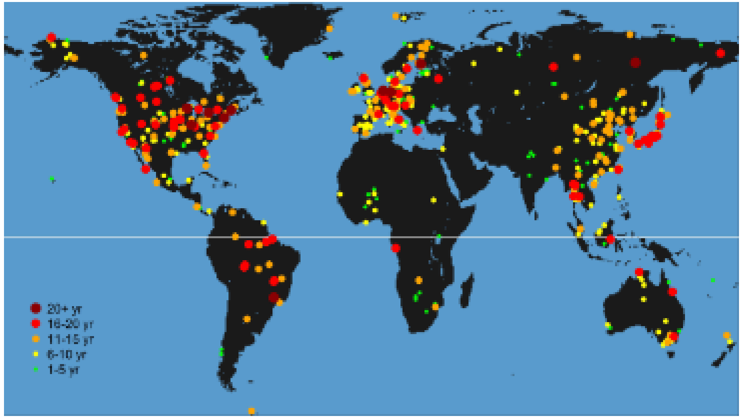
https://fluxnet.org/data/

FLUXNET

Micrometeorological network

FLUXNET is a global network of micrometeorological tower sites that use eddy covariance methods to measure the exchanges of carbon dioxide, water vapor, and energy between the biosphere and atmosphere...

FLUXNET 2015



PLEASE DO NOT PUBLISH ANYTHING USING THIS DATA WITHOUT REFERRING TO THEIR DATA POLICY

Met Office

Worth looking at work of

Boundary-Layer Meteorology (2023) 187:73–93
https://doi.org/10.1007/s10546-023-00784-3

RESEARCH ARTICLE

Check for updates

Evaluation of Surface Layer Stability Functions and Their Extension to First Order Turbulent Closures for Weakly and Strongly Stratified Stable Boundary Layer

Andrey V. Debolskiy[1,2,3] · Evgeny V. Mortikov[1,2,4] · Andrey V. Glazunov[1,2,4] · Christof Lüpkes[5]

Received: 1 September 2021 / Accepted: 9 January 2023 / Published online: 6 March 2023

Abstract

In this study, we utilize a generalization of Monin–Obukhov similarity theory to construct first order turbulent closures for single-column models of the atmospheric boundary layer (ABL). A set of widely used universal functions for dimensionless gradients is evaluated. Two test cases based on Large-Eddy Simulations (LES) experimental setups are considered – weakly stable ABL (GABLS1; Beare et al. in Bound Layer Meteorol 118(2):247–272, 2006), and very strongly stratified ABL (van der Linden et al. in Bound Layer Meteorol 173(2):165–192, 2019). The comparison shows that approximations obtained using a linear dimensionless velocity gradient tend to match the LES data more closely. In particular, the EFB (Energy- and Flux- Budget) closure proposed by Zilitinkevich et al. (Bound Layer Meteorol 146(3):341–373, 2013) has the best performance for the tests considered here. We also test surface layer "bulk formulas" based on these universal functions. The same LES data are utilized for comparison. The setup showcases the behavior of surface scheme, when one assumes that the velocity and temperature profiles in ABL are represented correctly. The advantages and disadvantages of different surface schemes are revealed.

Keywords Large-eddy simulation · Monin–Obukhov similarity theory · Stable boundary layer · Turbulence closures

Met Office

Inputs variables (features) in dataset

- LWdown,
- Precip,
- Psurf,
- Qair,
- SWdown,
- Tair,
- Ustar,
- Wind,
- Z0
- Leaf Area Index (maybe)

Output variable (target)

- stability function (phi)

Additional info: latitude, longitude (for data visualisation and rebalancing ONLY, not to be used as ML inputs).

Data file has been created by amalgamating hundreds of files and applying some quality control (thanks Helena Reid).

Initially ~17 millions samples, down to ~2.7 million when ensuring all potentially useful variables are available and pass QC on observed quantities

Initially data from 170 sites, now down to ~110.

Met Office

Suggested tasks

1. Powerpoint slides to explain the science behind what we are doing. Why it matters? What if affects? What does stable/unstable mean?

2. Slide explaining the input features: what are LWdown, Precip, Psurf, Qair, SWdown, Tair, Ustar, Wind, Z0, how are these measured?

3. Data exploration.
   - A Produce histogram of each input and output variables.
   - B Produce scatter plots of correlations between variables.
   - C Find max, min of all variables.
   - D Should some variables be looked at in terms of logs?
   - E Produce code to generate a normalised/standardised/rescaled data set.

4. Visualization: produce maps of total amount of data from each site. Maps of mean values of parameters at each site.

5. Start drafting a paper using Overleaf Latex. Think about literature review and sections and O(5) key figures.

---

6. Come up with method to do “leave one site out” during training and validation.

7. Assess how well balanced the data is.

8. Develop code to rebalance the dataset.

---

9. Train a random forest to predict stability function. Optimise hyper-parameters.

10. Train a MLP to predict the stability function. Optimise hyper-parameters.

Met Office

Aims for these 2 days

- Meet new people
- Do some communal work
- Make some progress on these tasks
- **Have fun !**