WRF Postprocessing

Torgeir Blæsterdalen

WindCoE, Department of Industrial Engineering
UiT-The Arctic University of Norway

May 3, 2018

Lecture topics

1. Reading NetCDF variables
2. Horizontal and vertical grids in WRF ARW
3. Distance between grid points and latitude, longitude
4. Calculating Temperature
5. Workflow for extracting and plotting WRF-data
6. Extracting output variables
7. Visualizing simulation data using MATLAB

The MATLAB version used for developing the example scripts for this lecture is 2016b.

1. Reading NetCDF variables I

As output-files from a WRF-simulation usually is quite big, it is necessary to extract only the variables of interest from a WRF-run. A simulation outfile from a domain with grid spacing of 2000 m $\times$ 2000 m is typically of the magnitude 50 GB. A description of the variables, variable name, its units, etc. can be found by typing the command ncdump -h <filename> in the terminal. In MATLAB, a NetCDF-file can be inspected using ncdisp. The command for extracting a NetCDF variable is ncread, e.g. the line

```
u  = ncread(<NetCDF-file>, 'U');
```

reads the variable U (x-wind component).
An important note is that the order of variable arguments might be different when examining files using ncdump or ncdisp, e.g.

ncdump: U(Time, bottom_top, south_north, west_east_stag)

ncdisp: U(west_east_stag, south_north, bottom_top, Time)

2. Horizontal and vertical grids in WRF ARW

Scalar variables like pressure, temperature, time, etc. are stored in the cell centers, i.e. in θ-coordinates whereas vector quantities like velocities and momentum variables are at the cell borders. The grids at the cell faces are referred to as *staggered grids*. This is shown in the figure below.

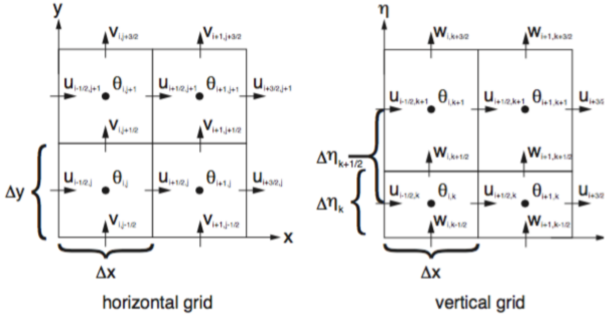


Figure: Grids of the WRF ARW, from Skamarock et al. [2008].

Staggered variables has to be *destaggered* if we're looking at the variable's value in the cell center. This can be done by computing the mean value, as shown in the example below. In this code snippet, θ-coordinate is computed for the East-West wind component in the middle of the simulation domain.

```
% ---- Destaggering the West - East wind component -------------
U  = ncread(fileID, 'U');

% Obtain two center points (icp = West-East, jcp = South-North)
icp = [round(size(U, 1)*.5) round(size(U, 1)*.5) + 1];
jcp = round(size(U, 2)*.5);

% Iterate over all timesteps at different eta-levels
u_theta = NaN(length(eta), size(U, 4));  % Preallocating

for t = 1:length(u)
   for z = 1:length(eta)
      u_theta(z, t) = 0.5*(U(icp(1), jcp, z, t) + U(icp(2), jcp,
          z, t));
```

```
  end
end

clear *cp t U z
```

Information of which variables are in staggered grid can be found using `ncdump` in the terminal or `ncdisp` in MATLAB.

3. Distance between grid points and latitude, longitude

A small excursion along the surface of sphere can be expressed

$$(\delta \boldsymbol{x}, \delta \boldsymbol{y}, \boldsymbol{z}) = (r\delta\lambda \cos\vartheta_o, r\delta\vartheta, z),$$

where r is the radius of the sphere, λ is the longitude, and ϑ is the latitude. The subscript o denotes the an arbitrary observational point. The excursion was done at constant altitude.
The distance from a point "O" to all WRF grid points can be expressed using the Pythagorean trigonometry

$$d(i,j) = \left(r^2 \cos^2\left(\frac{\vartheta_o + \vartheta(i,j)}{2}\right)\left(\lambda_o - \lambda(i,j)\right)^2 + r^2\left(\vartheta_o - \vartheta(i,j)\right)^2 \right)^{1/2}. \quad (1)$$

Here r is the radius of the Earth, the point "O" is at (λ_o, ϑ_o) and $(\lambda(i,j), \vartheta(i,j))$ denotes the WRF grid points.
A pseudo-code for extracting an example-variable at the grid point closest to the point "O" calculated using Eq 1 is given as a pseudo-code below.

```
% Read variable from WRF output file
variable = ncread('path to WRF output file', 'variable name');

% Longitude and latitude of the site
lon = 20.6804;
lat = 69.1867;

% Find the distance to all grid points
for longitudes && latitudes in variable
   d(i, j) = ...                                        (Eq.1)
end

% Find the minimum distance to the Rieppi site
[min_lon, min_lat] = find(minimum d(i,j));

save('Save variable to file')
```

4. Calculating Temperature

For calculating temperature at different heights, we have to derive it using potential temperature

$$\theta \equiv T\left(\frac{p_R}{p}\right)^{\kappa}, \tag{2}$$

where T is temperature, p is pressure, p_R is reference pressure and κ is the Poisson constant, defined by the gas constant R and the specific heat capacity at constant pressure, namely

$$\kappa \equiv \frac{R}{c_p}. \tag{3}$$

Including moisture in eq. 3, gives

$$\kappa = \frac{R_d(1-q) + R_w \cdot q}{c_{p,d}(1-q) + c_{p,w} \cdot q}, \tag{4}$$

where q is the water vapor mixing ratio and the subscripts d and w denotes dry and water, respectively. In WRF, some variables are described using base-state and perturbed quantities, these are denoted with an overline and apostrophe, respectively. Solving wrt. temperature, eq. 4 can now be rewritten to

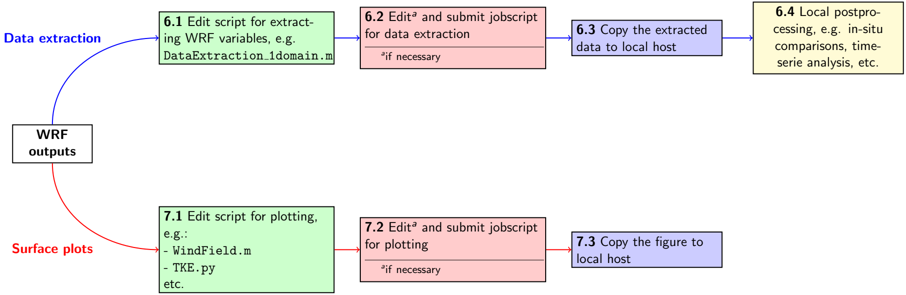
$$T(z) = (\overline{\theta} + \theta') \cdot \left(\frac{\overline{p(z)} + p'(z)}{p_R} \right)^{\frac{R_d(1-q(z))+R_w \cdot q(z)}{c_{p,d}(1-q(z))+c_{p,w} \cdot q(z)}}. \tag{5}$$

The constants used in the calculations are

$$R_d = 287 \text{ J kg}^{-1} \text{ K}^{-1}$$
$$R_w = 461.4 \text{ J kg}^{-1} \text{ K}^{-1}$$
$$c_{p,d} = \frac{7}{2} \cdot R_w$$
$$c_{p,w} = 4 \cdot R_w$$

5. Workflow for extracting and plotting WRF-data

Below is an example on how to extract and postprocess WRF data. Each of the boxes are more thoroughly described in the slide with the number corresponding to top left corner of the boxes.



6. Extracting output variables

6.1 Edit script for extracing WRF variables

The example-script `DataExtraction_1domain.m` extracts NetCFDF variables from WRF-runs at the centerpoint in the domain and saves the data in a MAT-file. The extracted variables are in θ-coordinates, i.e. they are unstaggered and only the first six η-values are harvested. All timesteps are retrieved.
Edit the file by entering

```
vi DataExtraction_1domain.m
```

Remember to change the simulation time in the script.

6.2 Edit and submit jobscript I

For running the data extraction script described in section 6.1, one can submit a jobscript like the JobscriptDataExtration_matlab.sh given below

```
#!/usr/bin/env bash
#:--------------------------------------------------------------------
# Jobscript for extracting data from WRF-simulations.
#
# Last edited: 02.Mayl.2018, Torgeir
#:--------------------------------------------------------------------

#SBATCH --job-name=Extracting_WRFdata
# Stallo account to charge
#SBATCH -A uit-hin-002

# Computation resources; nodes and cores
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --mem=30GB
```

6.2 Edit and submit jobscript II

```
# Runtime: d-hh:mm:ss (set a bit higher than expected)
#SBATCH --time 0-00:20:00

# Load relevant modules
module load netCDF/4.4.1.1-intel-2018a-HDF5-1.8.19
module load MATLAB/2015a-loc

# goes to the directory containing the script
cd ../PostProScripts/

% Run the script using MATLAB
matlab -nodisplay -nodesktop -r "DataExtraction_1domain"

exit 0
```

6.3 Copy the extracted data to local host

The output MAT-files from `DataExtraction_1domain.m` are in the magnitude 10 kB, and thus easy to download and work with on a local host.
Copying files from a remote to a local host can be done by

```
scp <stallo username>@stallo.uit.no:/<path to file>/<filename>
    <where to store file locally>
```

Example:

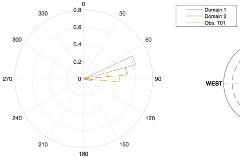
```
scp blasterdalen@stallo.uit.no:/home/blasterdalen/
    WRF_dataextracts/Simulation_12032014Domain2.mat .
```

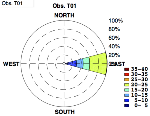
For Windows users, the program `WindSCP` can be used for copying/downloading files. The program can be downloaded for free from `https://winscp.net/eng/index.php`

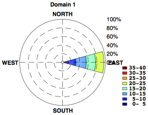
6.4 Local postprocessing I

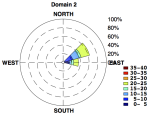
For postprocessing on a local host, the example script `WRFstats.m` is used. The script is reading variables from the MAT-files, and calculating wind direction, wind magnitude, temperature (according to Eq. 5). In-situ data from turbine T01 at Nygårdsfjellet is compared to the simulation data and correlations coefficients ρ and p-values are calculated. At the end of the script, timeseries comparing simulation data to the observational data are plotted. Example timeseries are shown below.

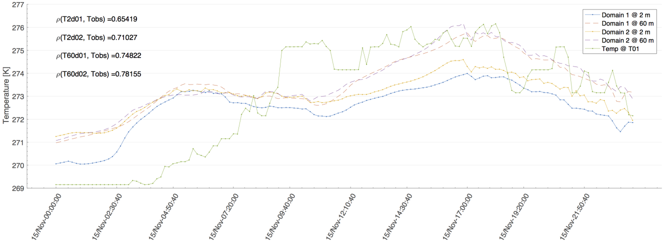
6.4 Local postprocessing II











7. Surface plots from WRF outputs I

7.1 Edit the script for plotting

Different scripts are made for plotting surface variables from WRF, for example `WindField_60m.py`, `SnowDebth.py`, `CloudCover.py` or `TKE.py`. The scripts are essential the same and in this section examples using the script `WindField.m` will be given.

The script `WindField.m` reads terrestrial data generated by `geogrid.exe` and wind data from WRF and plots elevation levels, wind arrows and wind magnitude as a surface. The script also requires the following libraries and functions:

- `m_map` : A very useful toolbox for dealing with projected data. The toolbox can be downloaded from `https://www.eoas.ubc.ca/~rich/map.html`
- `brewermap` Some additional (and nicer) colormaps to MATLAB's standard ones. From `https://se.mathworks.com/matlabcentral/fileexchange/45208-colorbrewer--attractive-and-distinctive-colo`

7. Surface plots from WRF outputs II

7.1 Edit the script for plotting

`save2pdf` Nice package for saving figures to PDF. From `https://se.mathworks.com/matlabcentral/fileexchange/16179-save2pdf`

Note that some of these add-ons are third party packages and is not developed by MathWorks.

Remember to declare the date, time and η-level of interest in the script before executing.

7.2 Edit and submit the jobscript I

For running the surface plotting script described in section 7.1, one can submit a jobscript like the `JobscriptPlotting_matlab.sh` given below

```
#!/usr/bin/env bash
#:-----------------------------------------------------------------
# Jobscript for extracting data from WRF-simulations.
#
# Last edited: 02.May.2018, Torgeir
#:-----------------------------------------------------------------

#SBATCH --job-name=MATLAB_plotting

# Stallo account to charge
#SBATCH -A uit-hin-002

# Computation resources; nodes and cores
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --mem=30GB
```

7.2 Edit and submit the jobscript II

```
# Runtime: d-hh:mm:ss (set a bit higher than expected)
#SBATCH --time 0-00:20:00

# Give priority to this job (requires shorter jobs than 4h, and
    is used for testing scripts)
# SBATCH --qos=devel

# Load relevant modules
module load netCDF/4.4.1.1-intel-2018a-HDF5-1.8.19
module load MATLAB/R2016a-intel-2016a

# Manouver to the directory containing the postprocessing script
cd ../PostProScripts/

matlab -nodisplay -nodesktop -r "WindField"

exit 0
```

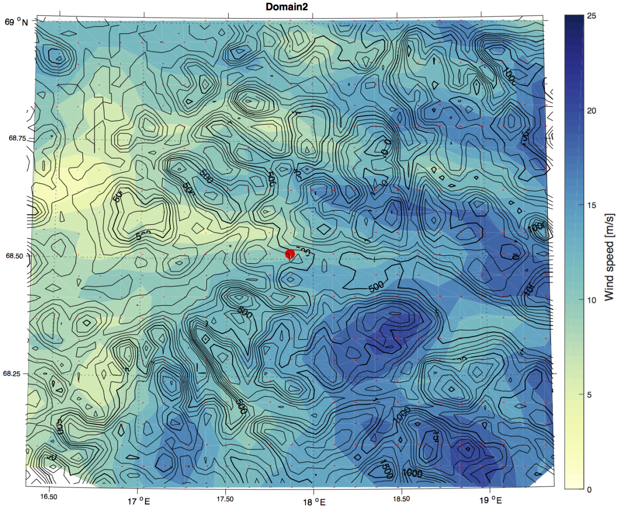
7.3 Copy the figure to local host I

As earlier, the figure can be downloaded to a local host using

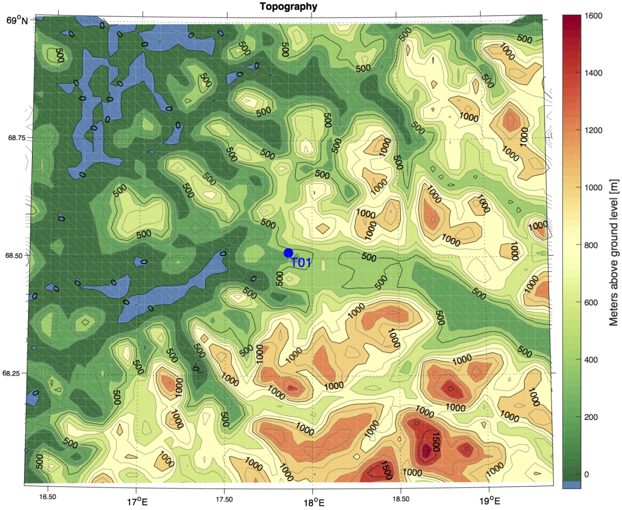
```
scp <username>@stallo.uit.no:/<path on Stallo>/<filename>
    /<localpath>/.
```

Below is a figure produced by the script `WindField.m`, where the reference turbine, T01, is labelled with a red dot in the middle of the frame. Elevation levels are drawn in black lines, with extra thick lines for every 500 meters. The last figure is a plot of the topography surrounding Nygårdsfjellet. This is produced by the script `TopographyNygaardsfjellet.m` and shows turbine T01 with a blue dot.

7.3 Copy the figure to local host II



7.3 Copy the figure to local host III



References

William C Skamarock, Joseph B Klemp, Jimy Dudhia, David O Gill, Dale M Barker, Wei Wang, and Jordan G Powers. A description of the advanced research wrf version 3. Technical report, DTIC Document, June 2008.