Preliminary Testing of Digital Filtering Initialization in AMPS

Michael G. Duda

MMM Division, NCAR

Boulder, Colorado, USA

The 3rd Antarctic Meteorological Observations, Modeling, and Forecasting Workshop
June 9-12, 2008 Madison, WI

A little background…

- Oct 2005 -- Parallel runs of WRF AMPS begin
  - Since the initial WRF AMPS implementation, various improvements have been made
- April 2008 -- WRF Version 3.0 is released
  - This release contains an implementation of a Digital Filtering Initialization (DFI)

Could the use of DFI in AMPS improve forecasts?

What is DFI?

Digital Filtering Initialization is a technique for removing *noise* from a model forecast by *initializing* the model analysis

- Noise is defined as high-frequency oscillations in the forecast
- This noise is caused by imbalances in interpolated initial fields
- Initialization modifies the model analysis state to eliminate noise

For high-resolution AMPS domains, we have good reason to suspect that interpolated fields will contain imbalances

- Terrain better resolved in AMPS than in GFS, etc.

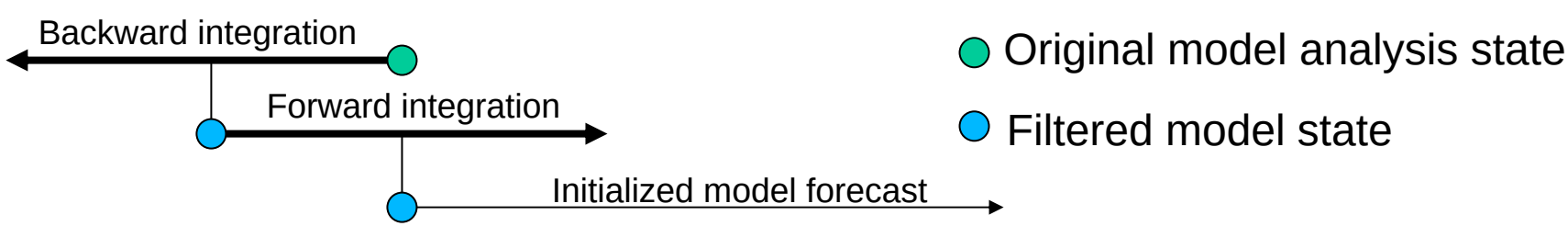
How does DFI work?

DFI applies a low-pass digital filter to time series of model fields, with the output of the filter used as the initialized state

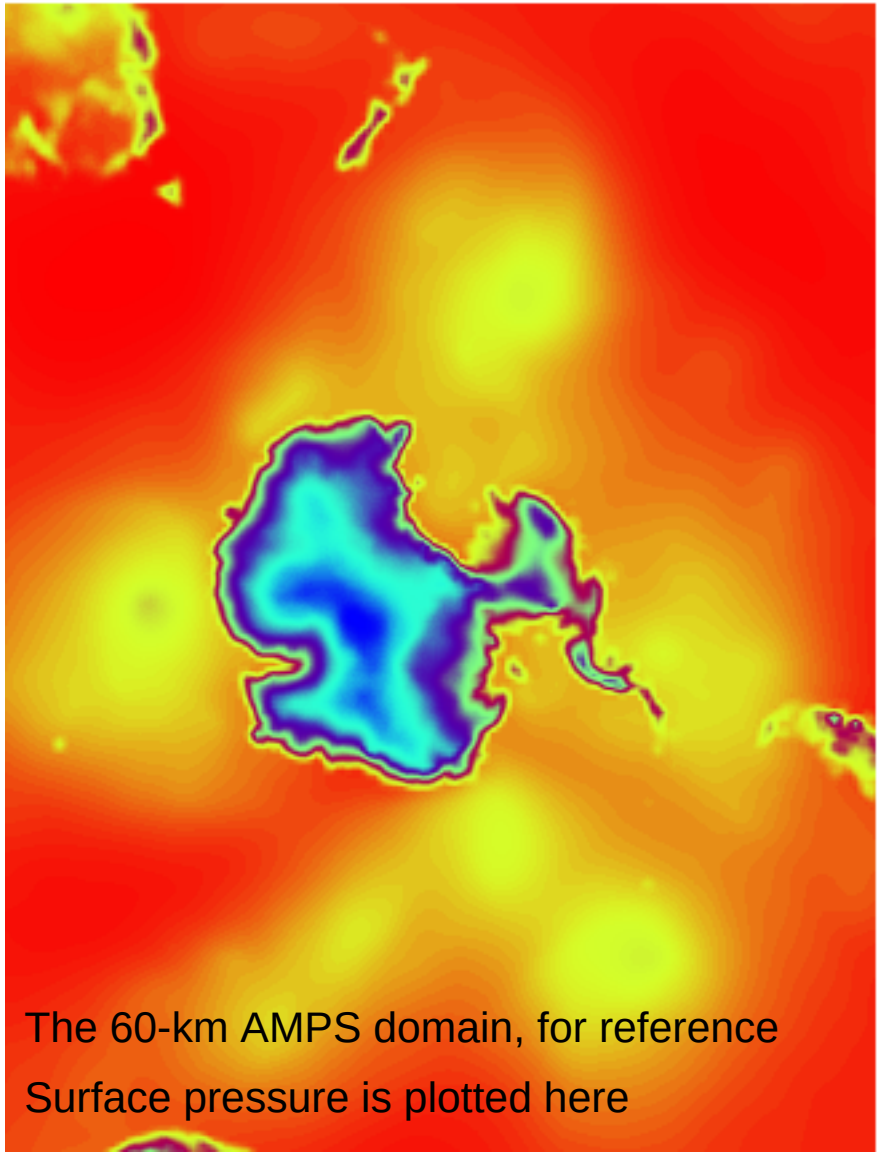
- Time series are produced by adiabatic, backward integration and diabatic, forward integration
- Each model grid point produces a time series for each variable
- The output of the filter valid at the analysis time provides a new, *initialized* model analysis state

Although DFI filters in time, *only the model analysis is modified by DFI*

We use the “Twice DFI” scheme in WRF:



DFI, Applied to AMPS 60-km Domain

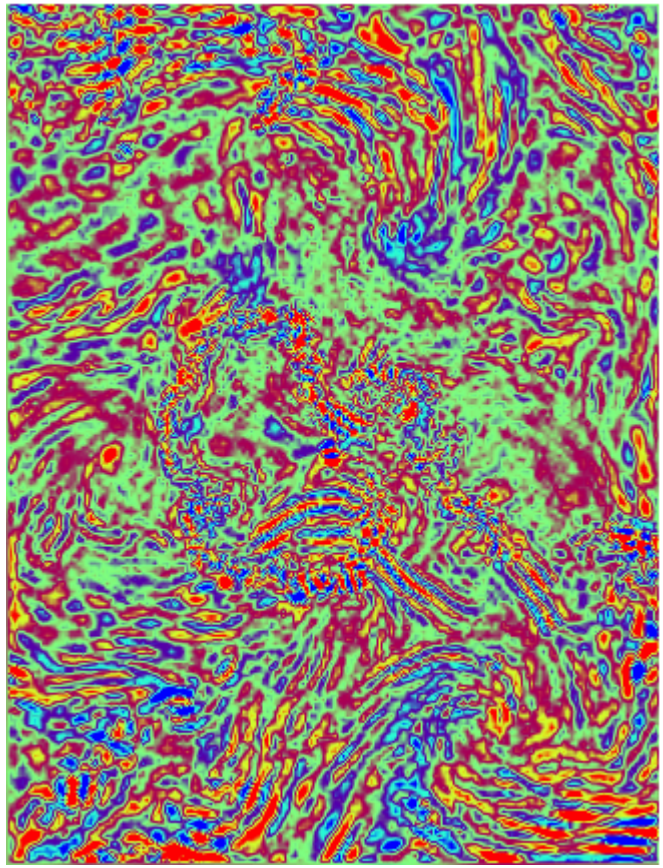


Noise throughout the domain is easily seen in the μ tendency field

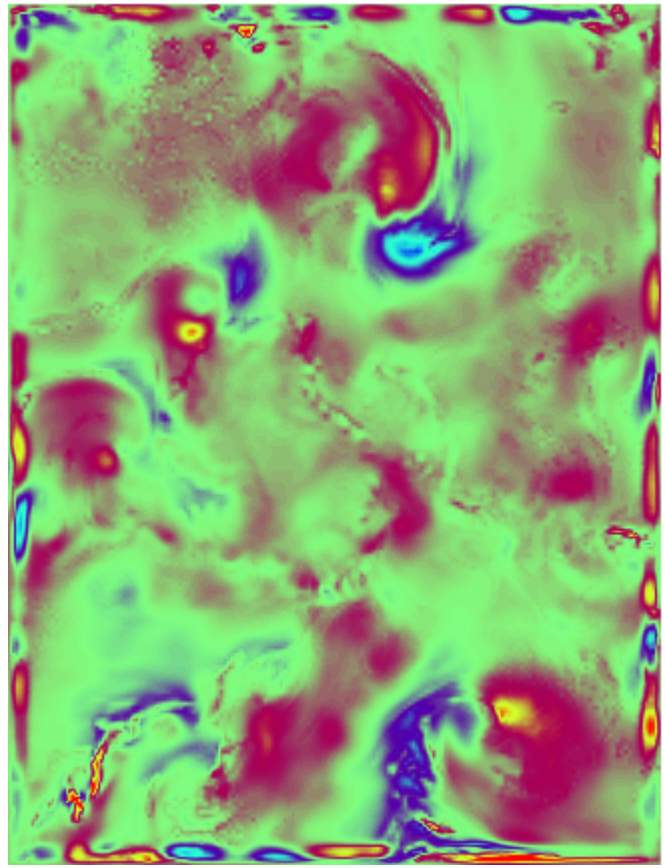
(μ = mass per unit area in a column of the model domain; used as a proxy for surface pressure)

DFI, Applied to AMPS 60-km Domain

Noise in the first 6 hours of a 60-km AMPS run, with and without DFI



$\frac{\partial \mu}{\partial t}$ from uninitialized forecast

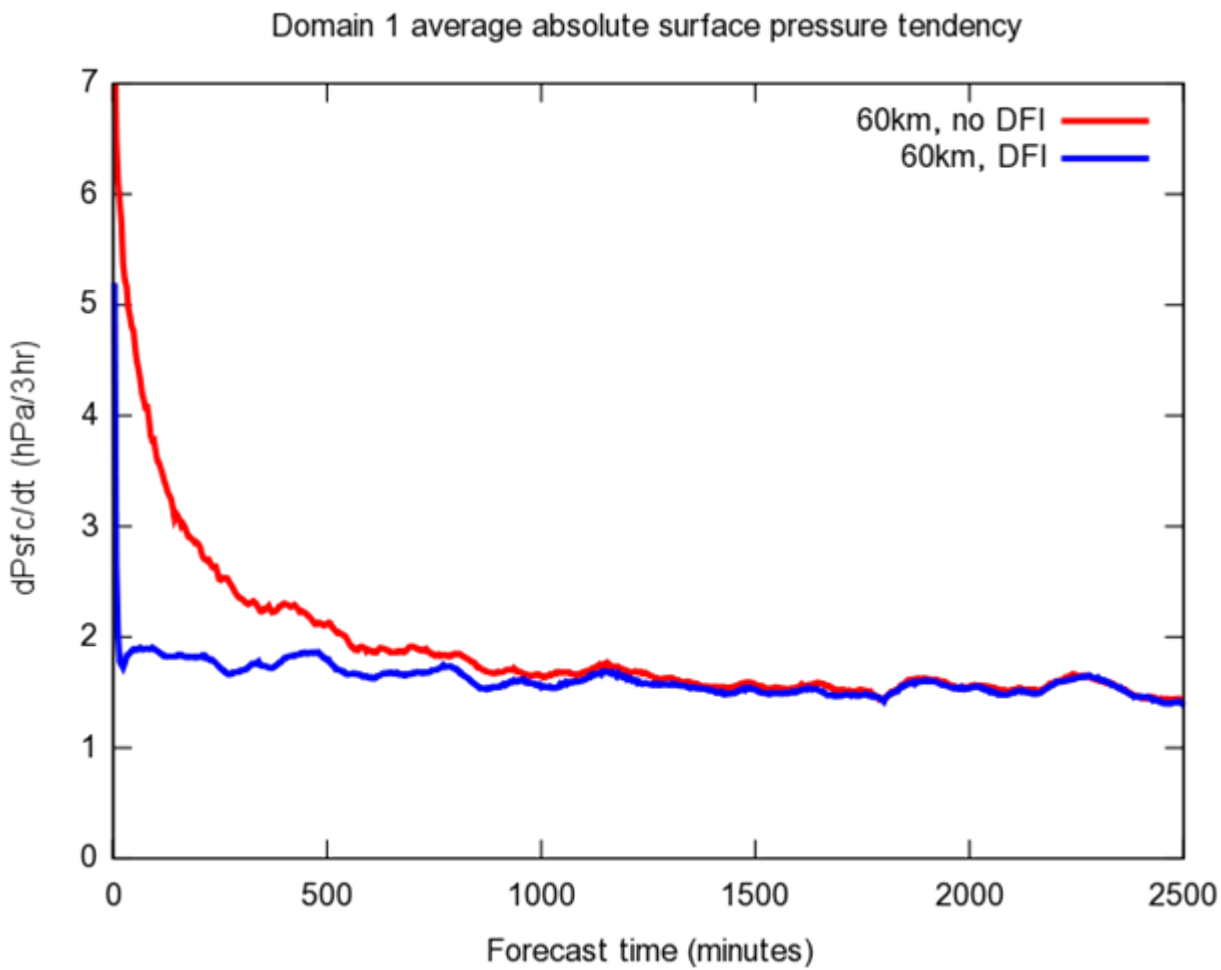


$\frac{\partial \mu}{\partial t}$ from forecast initialized with DFI

DFI, Applied to AMPS 60-km Domain

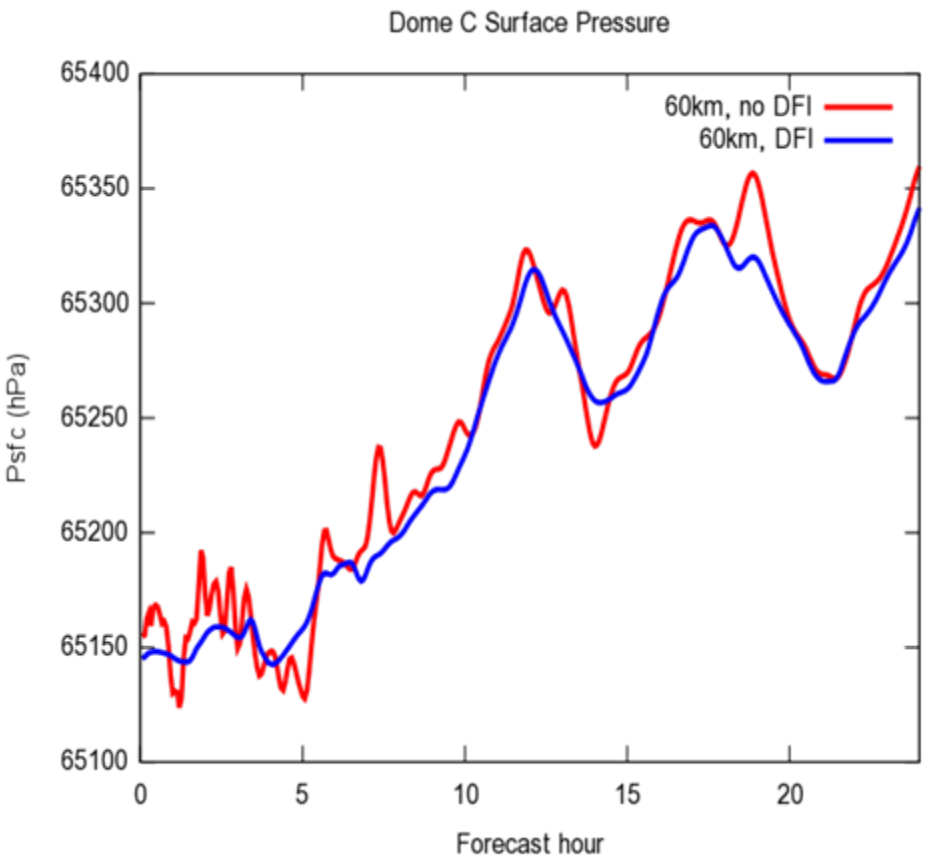
The amount of noise can be quantified as the domain-average absolute surface pressure tendency, for example

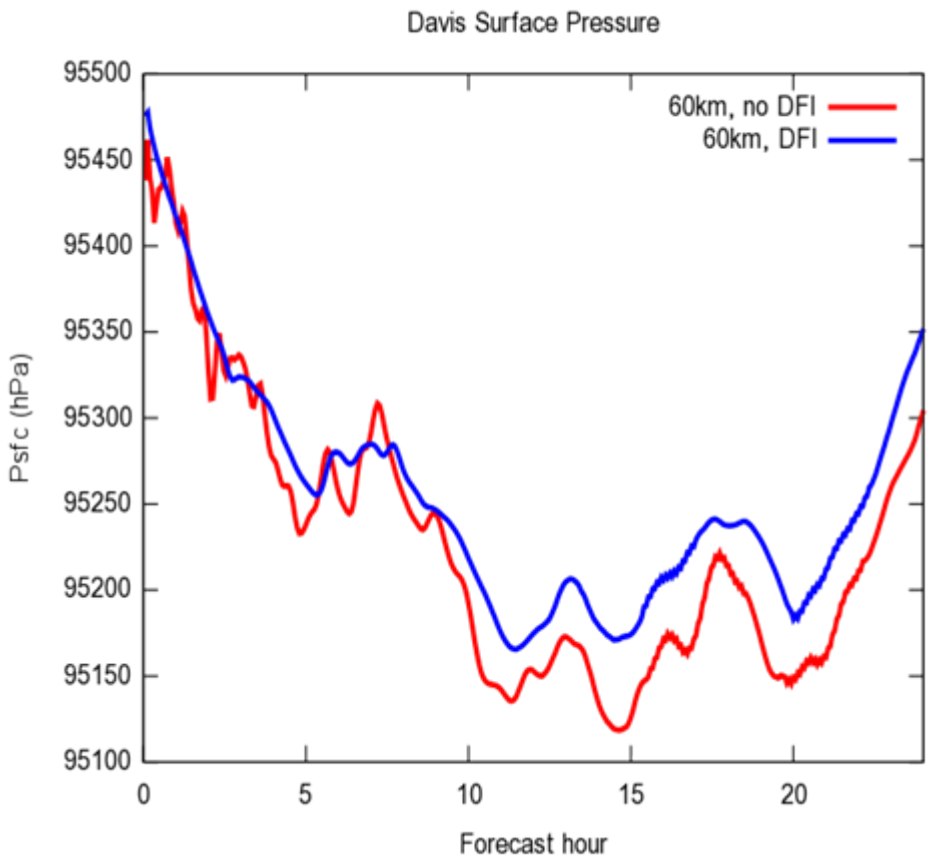
$$\frac{\sum_{i=1}^{ni}\sum_{j=1}^{nj}\left|\frac{\partial P_{sfc}}{\partial t}\right|}{ni \times nj}$$



DFI, Applied to AMPS 60-km Domain

The reduction in noise is also evident in time series from point locations within domain





But, what about nests?

AMPS has five two-way nested domains (in addition to coarse domain).

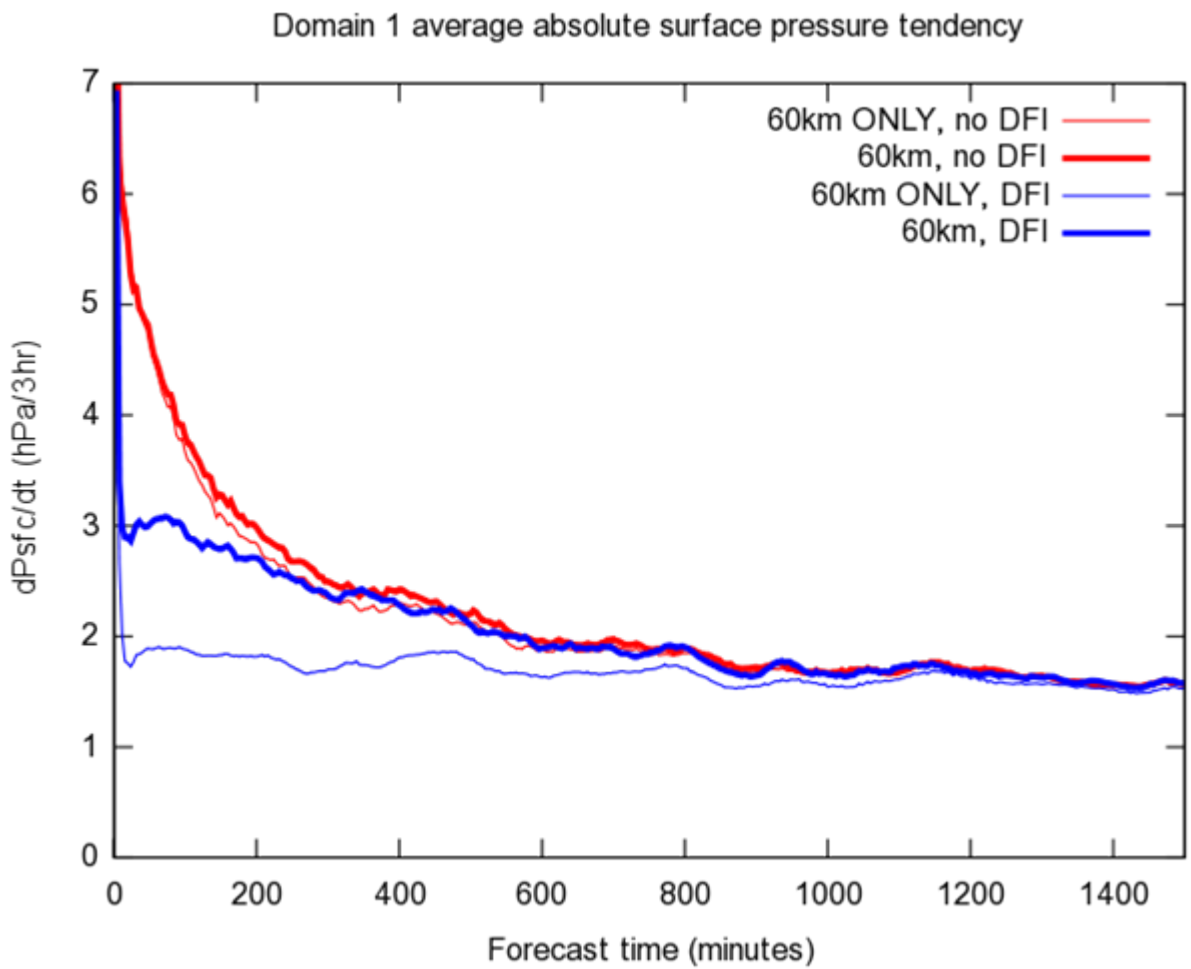
Current DFI implementation only handles one grid!

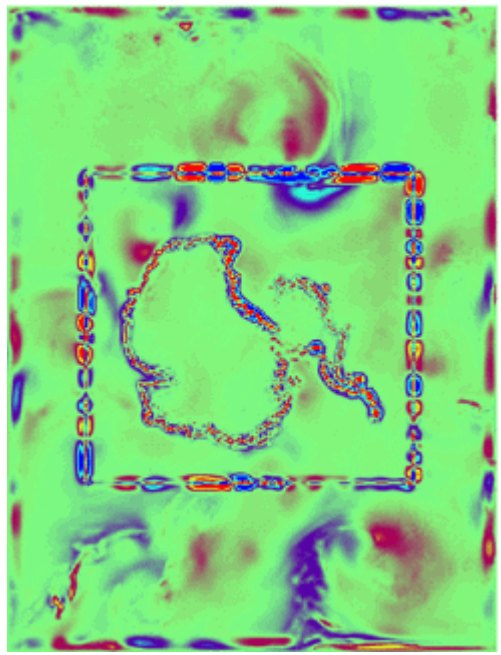
Our strategy:

1. Run DFI to initialize parent grid
2. Run short forecast from this initialized state to generate boundary conditions for nest
3. Run DFI to initialize nested grid
4. Launch nested WRF run using all initialized grids

DFI applied to 60/20-km domains

With a 20-km nest, noise level in initialized 60-km forecast is increased compared with initialized single-domain forecast





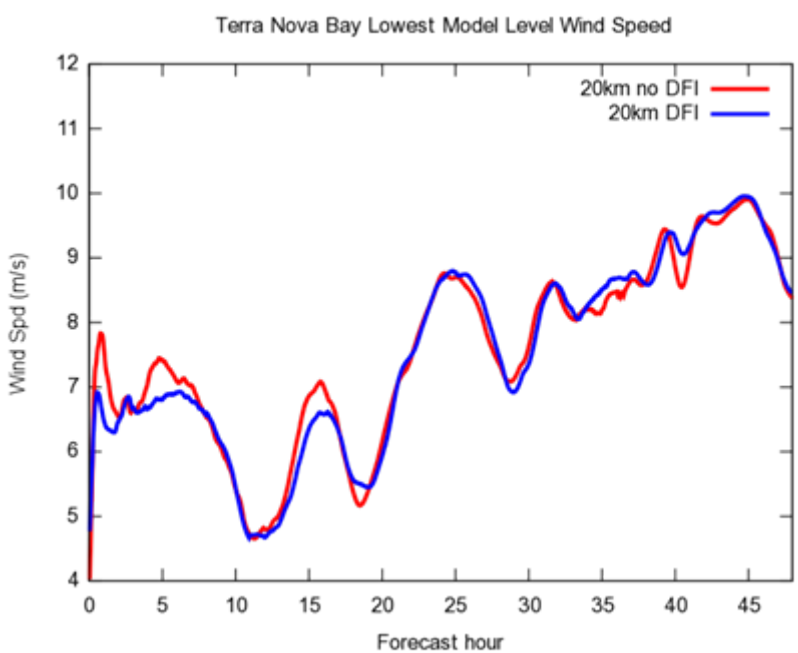
$\frac{\partial \mu}{\partial t}$ for 60-km AMPS grid

Impact of DFI on higher-resolution grids

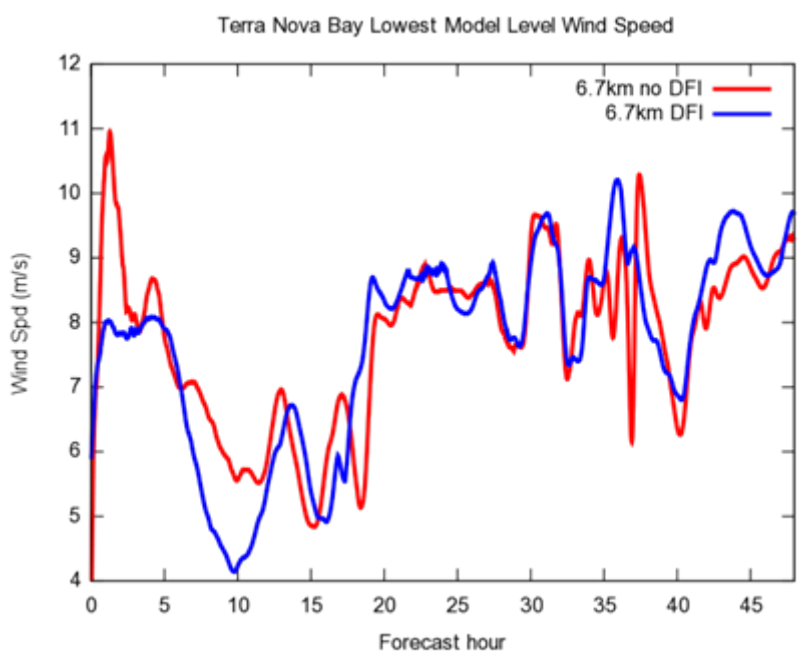
Apparently, DFI *can* have a more significant impact on forecasts for higher-resolution AMPS domains

- This could be caused by the better-resolved terrain in the nest, which could lead to larger initial imbalances

Wind speeds at Terra Nova Bay from 20km grid of 60/20 setup



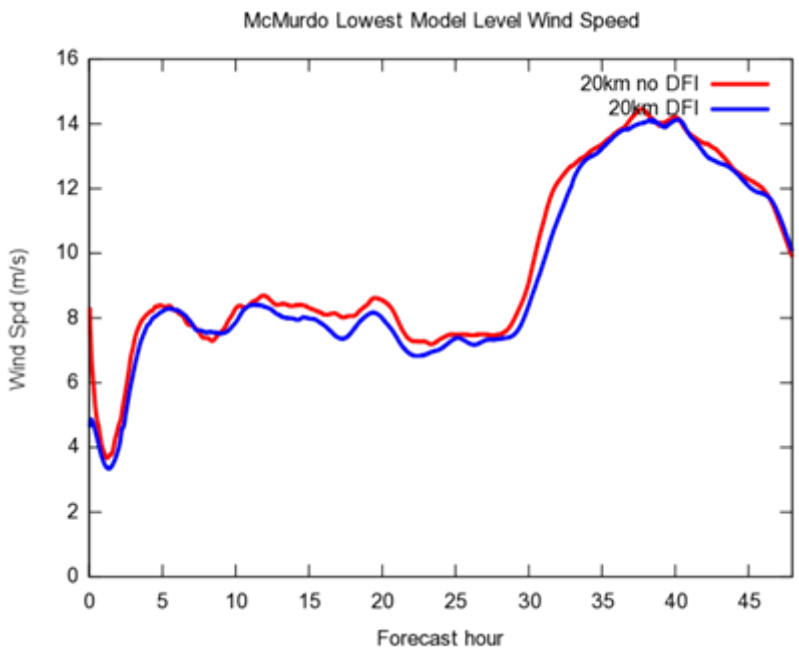
Wind speeds at Terra Nova Bay from 6.7km grid of 60/20/6.7 setup



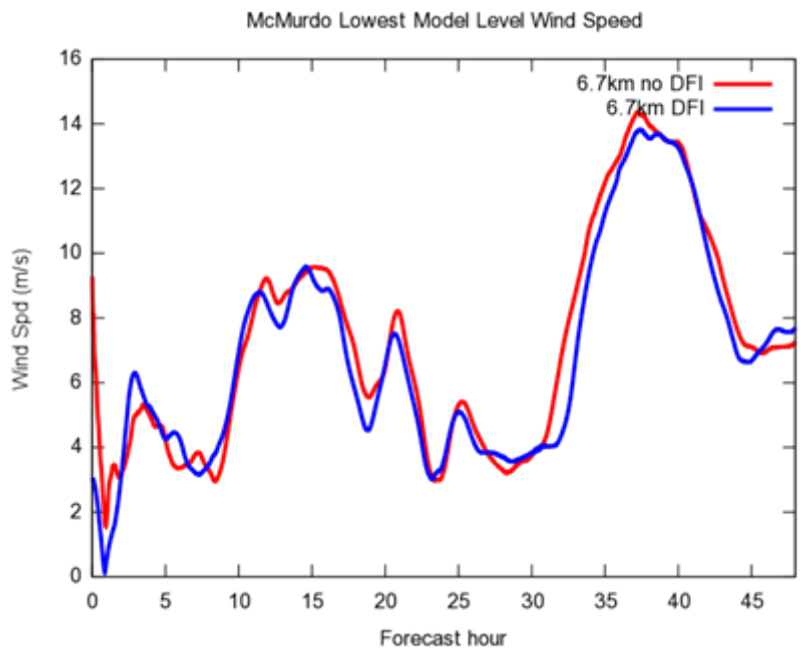
Impact of DFI on higher-resolution grids

In contrast to Terra Nova Bay, the differences in time series introduced by adding a nest might outweigh the changes due to DFI

Wind speeds at McMurdo from 20km grid of 60/20 setup



Wind speeds at McMurdo from 6.7km grid of 60/20/6.7 setup



Conclusions

- DFI does reduce noise
- Besides noise reduction, minor impact on forecasts
- Boundary conditions may be a weak point and could be improved
- To make proper use of DFI in AMPS, WRF DFI implementation should support nesting
  - This might cure the boundary problem for nests

Future Work

- Improve DFI implementation to work with nested domains concurrently
  - Could this help to solve noise problems at boundaries?
- Investigate the impact of noise on data assimilation
- Look for ways to reduce the computational cost of running DFI
  - Currently, we have used 3 hrs of backward integration plus 3 hrs of forward integration; potentially 6 hours of extra integration for all domains!

Questions?