# A fast converging and concise algorithm for computing the departure points in semi-Lagrangian weather and climate models

## EWGLAM workshop 2021
## 29/09/21

Michail Diamantakis and Filip Vana

ECMWF, Shinfield Park, Reading RG2 9AX, UK.

michail.diamantakis@ecmwf.int

# Motivation for this work

- **IFS semi-Lagrangian advection: trace backwards parcel trajectories from grid-points to find their departure points (DP) and interpolate advected fields to these DPs**
  - **Accurate** with long time-steps -> **Efficient**
  - **Multi-tracer efficient** (but non conserving)
- SETTLS (Hortal, 2001 QJRMS) is an iterative method to find DPs:
  1. *Current implementation in spherical coordinates is complicated!*
     - *Usual near pole curvature issues, rotate wind vectors to account for curvature*
     - *Solve in addition 3 nonlinear equations/grid point to find lat-lon of the DP*
     - *Some approximations needed to avoid division by 0*
     - *Not “friendly” for single-precision: keep the inversion procedure in double precision*
  2. *IFS long time-steps -> iterative method SETTLS converges slowly*
- New SETTLS for CY48R1 implemented fully on geocentric Cartesian system:
  - ***simplifies code greatly, better suited for single-precision, avoids pole problems***
  - ***Faster convergence (~ 40%) of DP calculation***

# Computing the departure points in IFS: SETTLS

**SETTLS** (Hortal, QJRMS 2002): 2nd order Taylor expansion of an arrival (grid) point position vector function at its departure point (DP)

Stable Extrapolation Two Time Level Scheme

$$r_a(t+\Delta t) = r_d(t) + \Delta t \cdot \left(\frac{Dr}{Dt}\right)_d^t + \frac{\Delta t^2}{2} \cdot \left(\frac{D^2 r}{Dt^2}\right)_{AV}$$

AV: average value along SL trajectory

$$\left(\frac{Dr}{Dt}\right)_d^t = V_d(t), \quad \left(\frac{D^2 r}{Dt^2}\right)_{AV} = \left(\frac{DV}{Dt}\right)_{AV} \approx \frac{V_a(t) - V_d(t-\Delta t)}{\Delta t}$$

Hence,

$$r_a(t+\Delta t) \approx r_d(t) + \frac{\Delta t}{2} \cdot \left(V_a(t) + \{2V(t) - V(t-\Delta t)\}_d\right)$$

Therefore DP can be computed by iterative sequence:

$$r_d^{(0)} = r_a - \Delta t V(r_a, t)$$

$$r_d^{(k)} = r_a - \frac{\Delta t}{2} \cdot \left(V_a(t) + \{2V(t) - V(t-\Delta t)\}\Big|_{r_d^{(k-1)}}\right) \quad k = 1, 2, \ldots K$$

Interpolate at $r_d^{(k-1)}$ In the IFS: K=5

- Convergence of iterative scheme must satisfy Lipschitz condition $\Delta t \left|\partial V / \partial r\right| < 1$
- Implementation on the sphere results in a more complex algorithm

# On the sphere: compute DP lat/lon  [from Temperton et al, QJRMS 2001]

We use the fact that $\phi$ is small to approximate $\sin\phi$ and $\cos\phi$, thereby avoiding a potential division by zero:

$$\sin\phi \approx \phi\left(1-\frac{1}{6}\phi^2\right) = |\mathbf{v}_\mathrm{M}|\frac{\Delta t}{a}\left(1-\frac{1}{6}\phi^2\right),$$

$$\cos\phi \approx 1-\frac{1}{2}\phi^2.$$

Thus the iterative scheme becomes:

$$\mathbf{r}_\mathrm{M}^{(k+1)} = \mathbf{r}_\mathrm{A}\{1-\tfrac{1}{2}(\phi^{(k)})^2\} - \mathfrak{R}(\mathbf{v}_\mathrm{M}^{(k)})\Delta t\{1-\tfrac{1}{6}(\phi^{(k)})^2\}.$$

Defining $(u', v')^T = \mathfrak{R}(\mathbf{v}_\mathrm{M}^{(k)})$ and resolving in Cartesian $(x, y, z)$ coordinates, we obtain:

$$\sin\theta_\mathrm{M}^{(k+1)} = \sin\theta_\mathrm{A}\left\{1-\frac{1}{2}(\phi^{(k)})^2\right\} - v'\frac{\Delta t}{a}\cos\theta_\mathrm{A}\left\{1-\frac{1}{6}(\phi^{(k)})^2\right\},$$

$$\cos\theta_\mathrm{M}^{(k+1)}\cos(\lambda_\mathrm{M}^{(k+1)}-\lambda_\mathrm{A}) = \cos\theta_\mathrm{A}\left\{1-\frac{1}{2}(\phi^{(k)})^2\right\} + \frac{v'\Delta t}{a}\sin\theta_\mathrm{A}\left\{1-\frac{1}{6}(\phi^{(k)})^2\right\},$$

$$\cos\theta_\mathrm{M}^{(k+1)}\sin(\lambda_\mathrm{M}^{(k+1)}-\lambda_\mathrm{A}) = \frac{-u'\Delta t}{a}\left\{1-\frac{1}{6}(\phi^{(k)})^2\right\}.$$

$$\sin\theta_\mathrm{D} = \sin\theta_\mathrm{A}\cos 2\phi - 2\cos\phi\frac{v'\Delta t}{a}\cos\theta_\mathrm{A}\left(1-\frac{1}{6}\phi^2\right),$$

$$\cos\theta_\mathrm{D}\cos(\lambda_\mathrm{D}-\lambda_\mathrm{A}) = \cos\theta_\mathrm{A}\cos 2\phi + 2\cos\phi\frac{v'\Delta t}{a}\sin\theta_\mathrm{A}\left(1-\frac{1}{6}\phi^2\right)$$

$$\cos\theta_\mathrm{D}\sin(\lambda_\mathrm{D}-\lambda_\mathrm{A}) = -2\cos\phi\frac{u'\Delta t}{a}\left(1-\frac{1}{6}\phi^2\right),$$

- Estimate angle φ between DP and gridpoint position vector
- Approximate sinφ, cosφ

- Iterate nonlinear products
- At each iteration apply rotation matrix $\mathfrak{R}(\mathbf{v}_\mathrm{M}^{(k)})$

(iterative equations copied from the Temperton et al paper and refer to old mid-point iterative scheme but very similar for SETTLS)

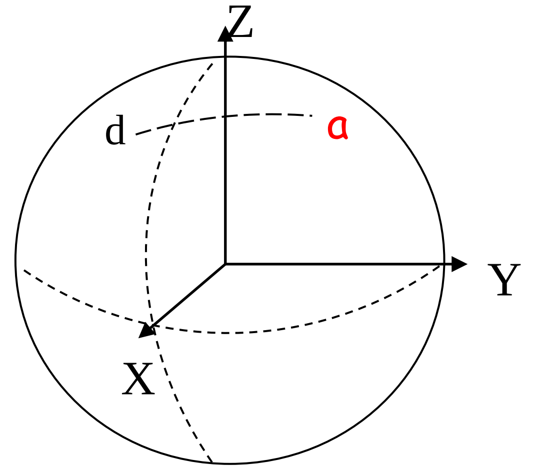
After iterations invert at each gridpoint a system of 3 nonlinear equations to obtain the lat/lon of the DP $\theta_\mathrm{D}$, $\lambda_\mathrm{D}$

# SETTLS iterations on the sphere using a geocentric Cartesian system

- Introduce Cartesian geocentric coordinates XYZ as in Ritchie MWR87 and project $(u, v) \rightarrow (U, V, W)$

$$X_j = a\cos\theta_j \cos\lambda_j, \quad Y_j = a\cos\theta_j \sin\lambda_j, \quad Z_j = a\sin\theta_j, \quad \Rightarrow \quad U_j = -u_j \sin\lambda_j - v_j \sin\theta_j \cos\lambda_j, \quad V_j = u_j\cos\lambda_j - v_j \sin\lambda_j \sin\theta_j, \quad W_j = v_j \cos\theta_j$$

η: pressure based terrain following vertical coord




- SETTLS on sphere iterations for DP d using XYZη coords:

$\ell = 1, 2, \ldots$

$$X_d^{(\ell)} = X_j - \frac{\Delta t}{2}\left[U_j^t + \left(2U^t - U^{t-\Delta t}\right)_{d(\ell-1)}\right]$$

$$Y_d^{(\ell)} = Y_j - \frac{\Delta t}{2}\left[V_j^t + \left(2V^t - V^{t-\Delta t}\right)_{d(\ell-1)}\right]$$

$$Z_d^{(\ell)} = Z_j - \frac{\Delta t}{2}\left[W_j^t + \left(2W^t - W^{t-\Delta t}\right)_{d(\ell-1)}\right]$$

$$\eta_d^{(\ell)} = \eta_j - \frac{\Delta t}{2}\left[\dot\eta_j^t + \left(2\dot\eta^t - \dot\eta^{t-\Delta t}\right)_{d(\ell-1)}\right]$$

$$\lambda_d^{(\ell)} = \mathrm{ATAN2}(Y_d^{(\ell)}, X_d^{(\ell)})$$

$$\theta_d^{(\ell)} = \arcsin \frac{Z_d^{(\ell)}}{\sqrt{X_d^{(\ell)2} + Y_d^{(\ell)2} + Z_d^{(\ell)2}}}$$

with

$$X_d^{(0)} = X_j - \Delta t U_j^t$$

$$Y_d^{(0)} = Y_j - \Delta t V_j^t$$

$$Z_d^{(0)} = Z_j - \Delta t W_j^t$$

$$\eta_d^{(0)} = \eta_j - \Delta t \dot\eta_j^t.$$

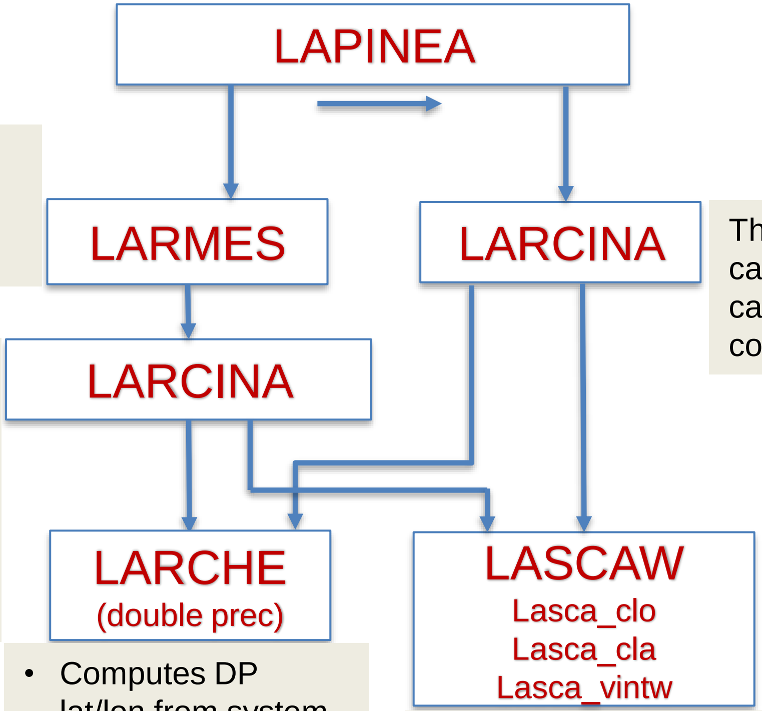
Start iterations with 1st order scheme

- Updating $\lambda_d, \theta_d$ greatly simplified
- No approximations
- No rotation matrix
- All calculations in single precision
- Avoids pole issues
- Very compact code
- Extra linear interpolation for W but interpolation weight available (fast)

# More compact code: fewer subroutines + fewer lines of code

Iterates to compute vertical DP and trig products depending on lat/lon of the DP

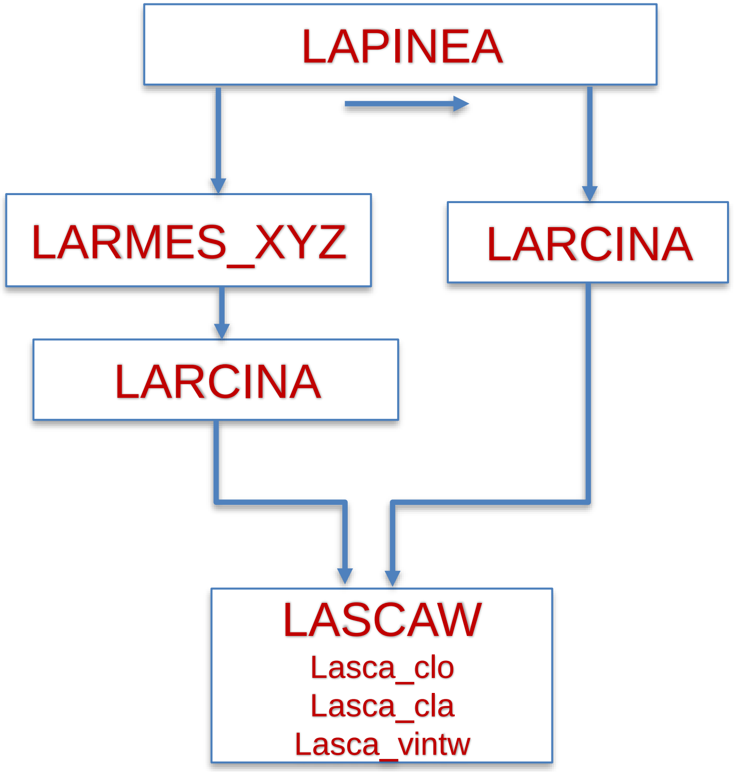
Interface sub to compute interpolation weights and finalise lat/lon of the DP. Furthermore, when called by LARMES, interpolates wind components at the DP (needed by iterative scheme)




This is the final call after DP calculation is complete

- Computes DP lat/lon from system of nonlinear trig equations
- Computes "rotation matrix" needed for momentum component interpolation to the DP

Computes trilinear, tricubic, bicubic-quintic weights needed to interpolate GFL, GMV vars at the DP. Same weights for all advected vars: efficient!



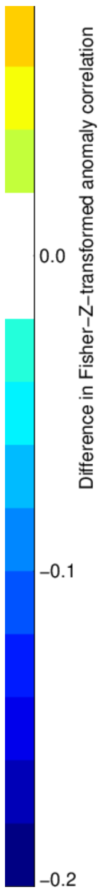

Cartesian SETTLS scheme: compact readable code + faster convergence

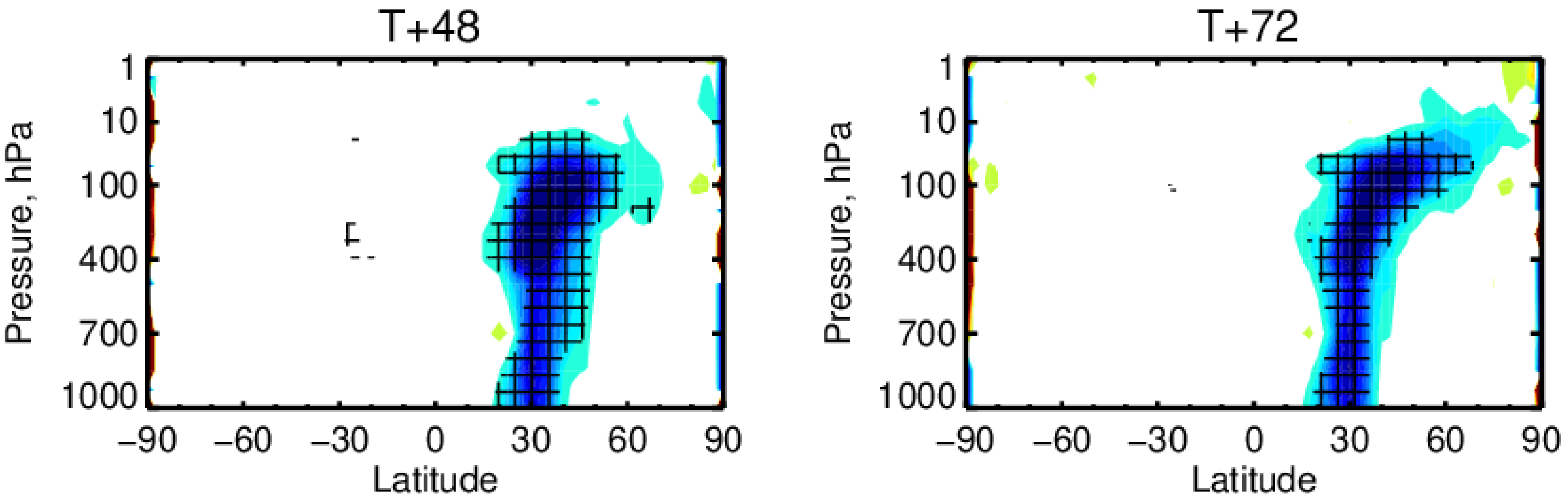
# Converging the DP iterations is important for forecast accuracy

- DP iterations converge slowly in IFS: rate depends on the deformational (Lipschitz) Courant number
- Lack of sufficient convergence: skill degradation and TC performance degradation (see Diamantakis & Magnusson MWR 2016)
- From CY41R2: 5 iterations are used compared with 3 in previous cycles

Evolution of timestep/grid spacing in IFS history …

| Horizontal Res | Vertical levs | tstep (s) | $\Delta t/\Delta x$ |
|---|---|---|---|
| TL399 (50km) | 91 | 1200 | 0.02 |
| TL511 (40km) | 91 | 900 | 0.02 |
| TL1279 (16km) | 91 | 600 | 0.04 |
| Tco1279 (9km) | 137 | 450 | 0.05 |




geopotential ACC reduction with 3 versus 5 iterations

BLUE worse

# New SETTLS algorithm with faster convergence (saves 2 iterations/step)

– Pre-compute $X_j, Y_j, Z_j$ at each grid-point $j$

– At each time-step, obtain $U_j, V_j, W_j$ from $u_j, v_j$

**if** (step count $> 0$) **then**

Initialize the horizontal components of the DP from the previous time-step:

$$\lambda_d^{(0)} = \lambda_d^{(K)}\Big|_{t-\Delta t}, \quad \theta_d^{(0)} = \theta_d^{(K)}\Big|_{t-\Delta t}$$

**if** $(\dot\eta_j^t \cdot \dot\eta_j^{t-\Delta t} \geq 0)$ **then**

$$\eta_d^{(0)} = \eta_d^{(K)}\Big|_{t-\Delta t}$$

**else**

$$\eta_d^{(0)} = \eta_j - \Delta t \dot\eta_j^t$$

**end if**

New starting values procedure:

- Use previous timestep DP to start iterations
- BUT: when vertical wind direction changes in two consecutive steps then don't trust the old value to start iterations, better switch to 1st order Euler predictor

**else**

compute starting values for the DP iterations using (13)-(16)

**end if**

**for** $\ell = 1, ..., K$ **do**

– Interpolate $U, V, W, \dot\eta$ to available estimate for the DP $d^{(\ell-1)}$

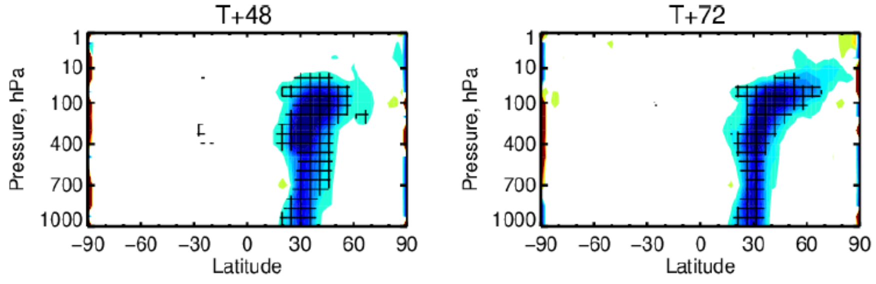
– Compute $X_d,\ Y_d,\ Z_d,\ \eta_d$ DP :

– Compute corresponding $\lambda_d, \theta_d$

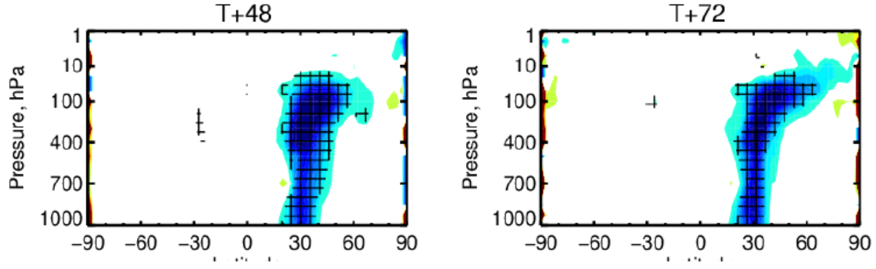
**end for**

Save DP $d^{(K)} \equiv (\lambda_{d^{(K)}}, \theta_{d^{(K)}}, \eta_{d^{(K)}})$ to be used as a predictor at next time step DP iterations

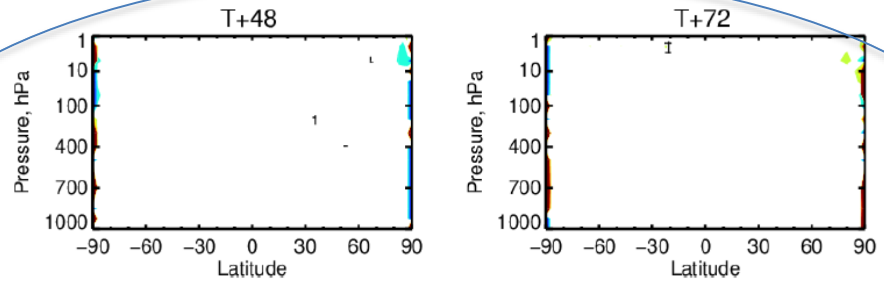
# Validation of new faster SETTLS against oper forecast at 9km



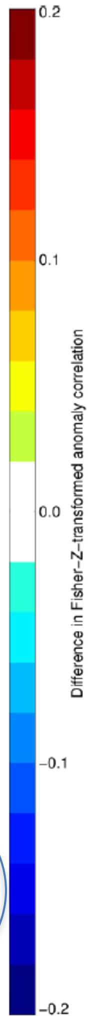

(a) operational SETTLS with 3 iterations

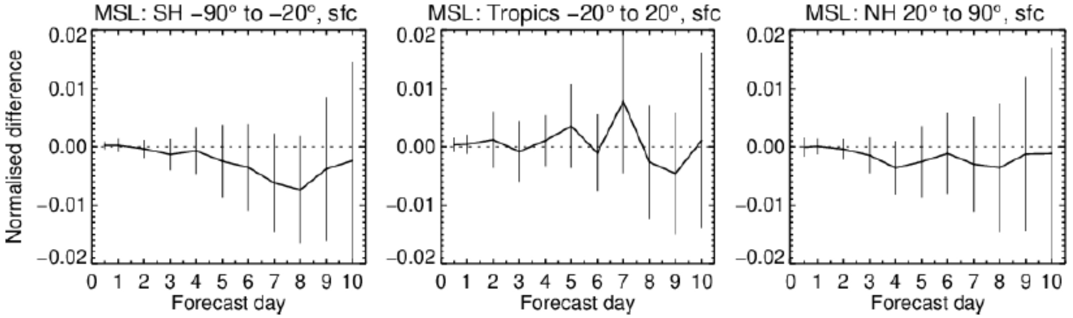



(b) new Cartesian SETTLS with 3 iterations

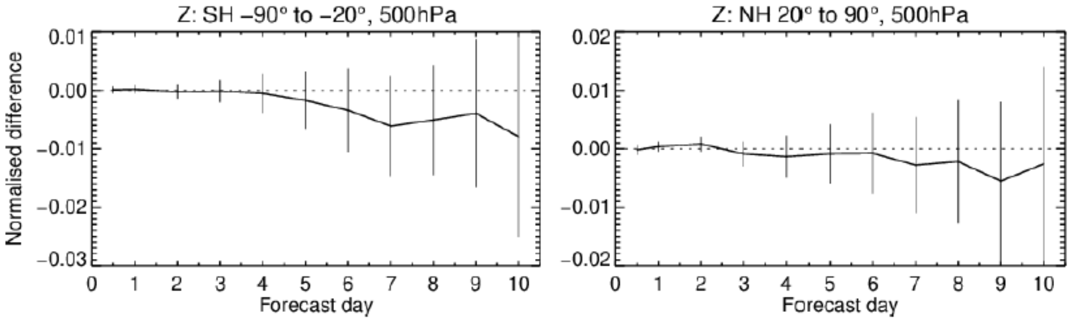



(c) New Cartesian SETTLS with 3 iterations and predictor for the departure points




(a) MSL RMSE scaled difference (negative better)




(b) Geopotential 500 hPa RMSE scaled difference (negative better)

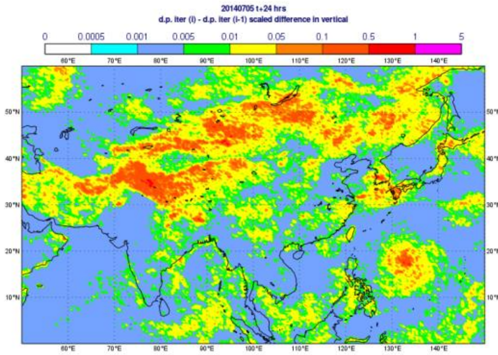
4 months in total 2+2 winter/summer: same accuracy with operational scheme

Cartesian SETTLS with new starting values equally accurate with operational scheme. Blue is worse (reduction of ACC)

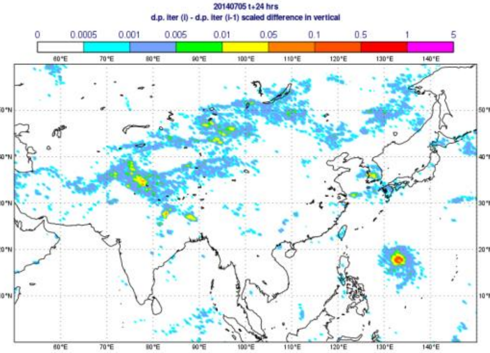
# Convergence diagnostics from typhoon Neoguri case

$$\frac{\left|\eta_d^{(2)} - \eta_d^{(1)}\right|}{\Delta\eta}$$

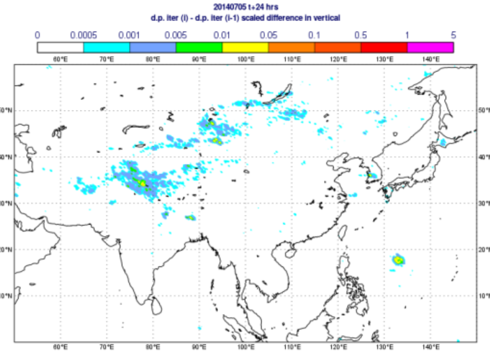
$\Delta\eta$ : vertical resolution at corresponding arrival(grid)point




$$\frac{\left|\eta_d^{(3)} - \eta_d^{(2)}\right|}{\Delta\eta}$$



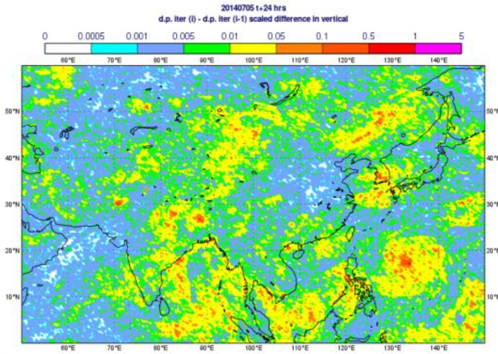

$$\frac{\left|\eta_d^{(4)} - \eta_d^{(3)}\right|}{\Delta\eta}$$

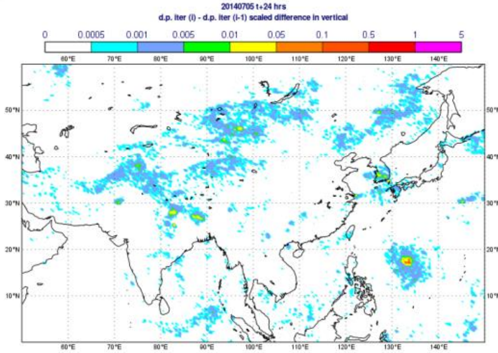


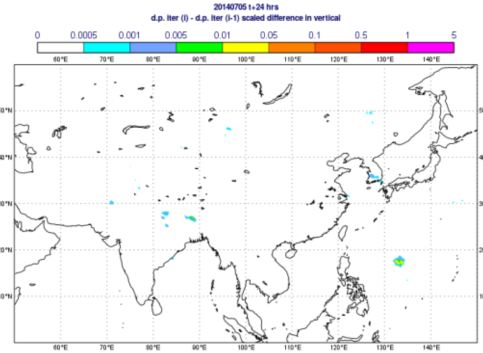

Old predictor

- Faster convergence of new algorithm
- New predictor gives better starting values
- Has converged by 3rd iteration

TC case from MWR2016, Diamantakis & Magnusson

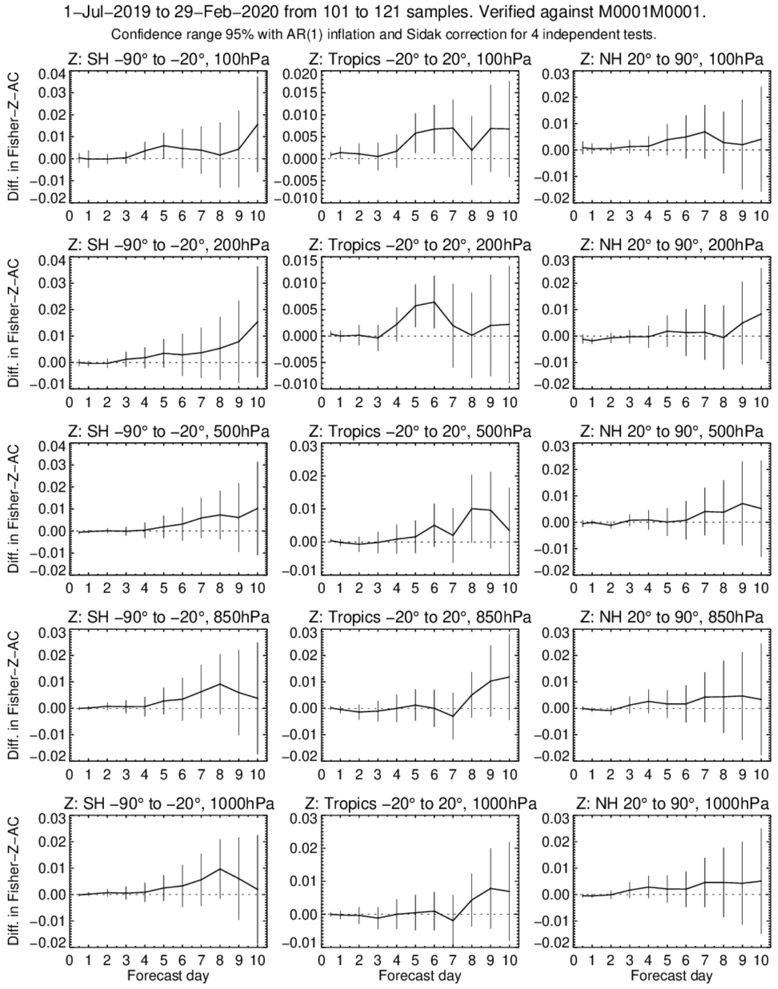


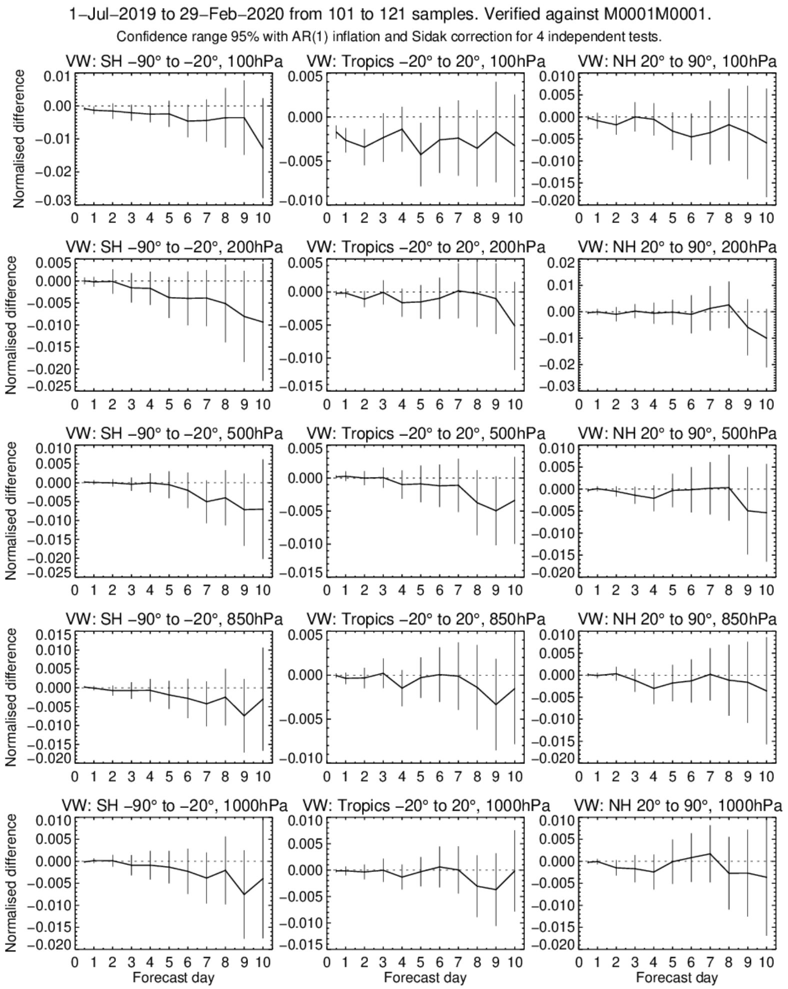



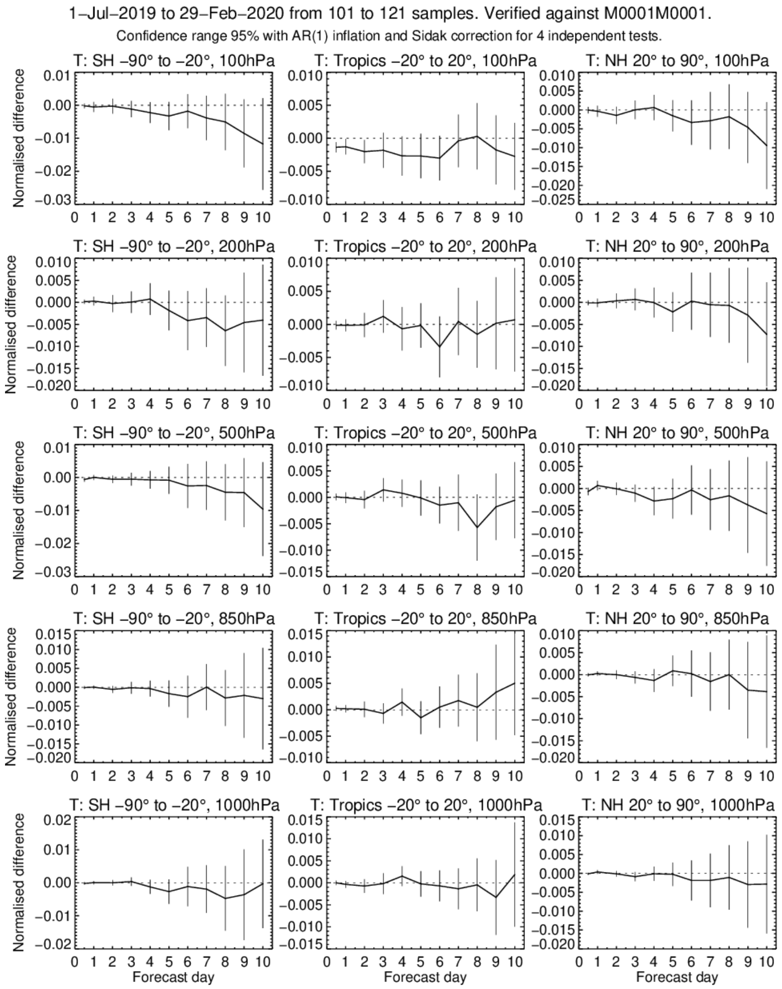




New predictor

# Combined winter/summer Tco1279 scores for Z, VW, T (4 months)



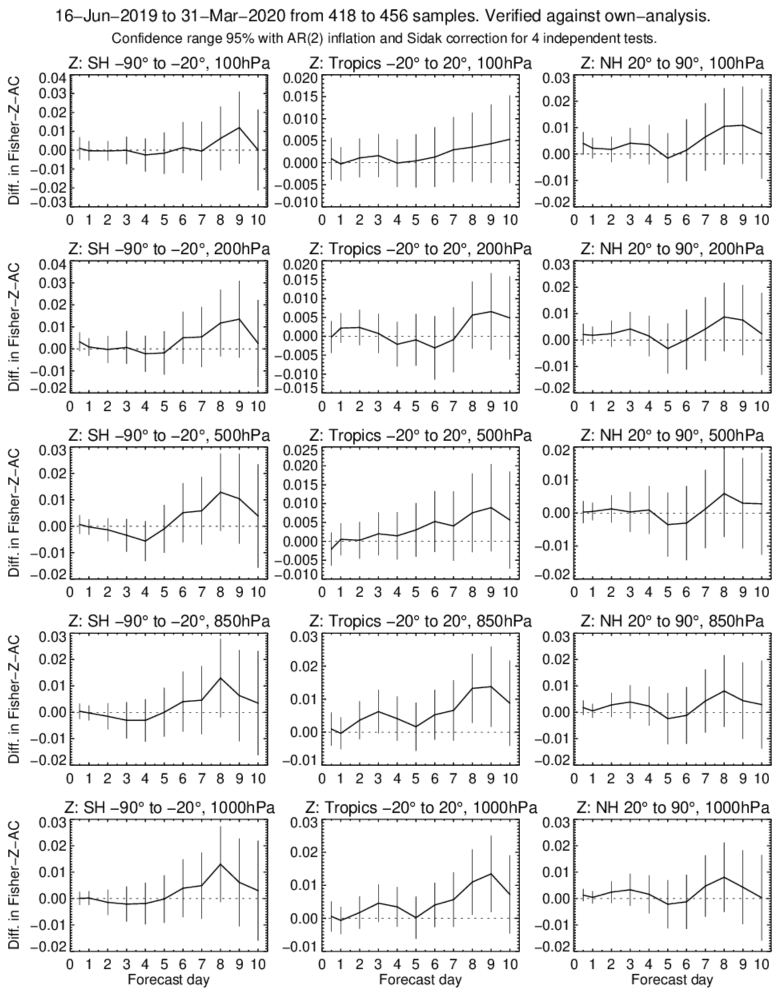

MhhjvMhhho – MhcsoMhheo

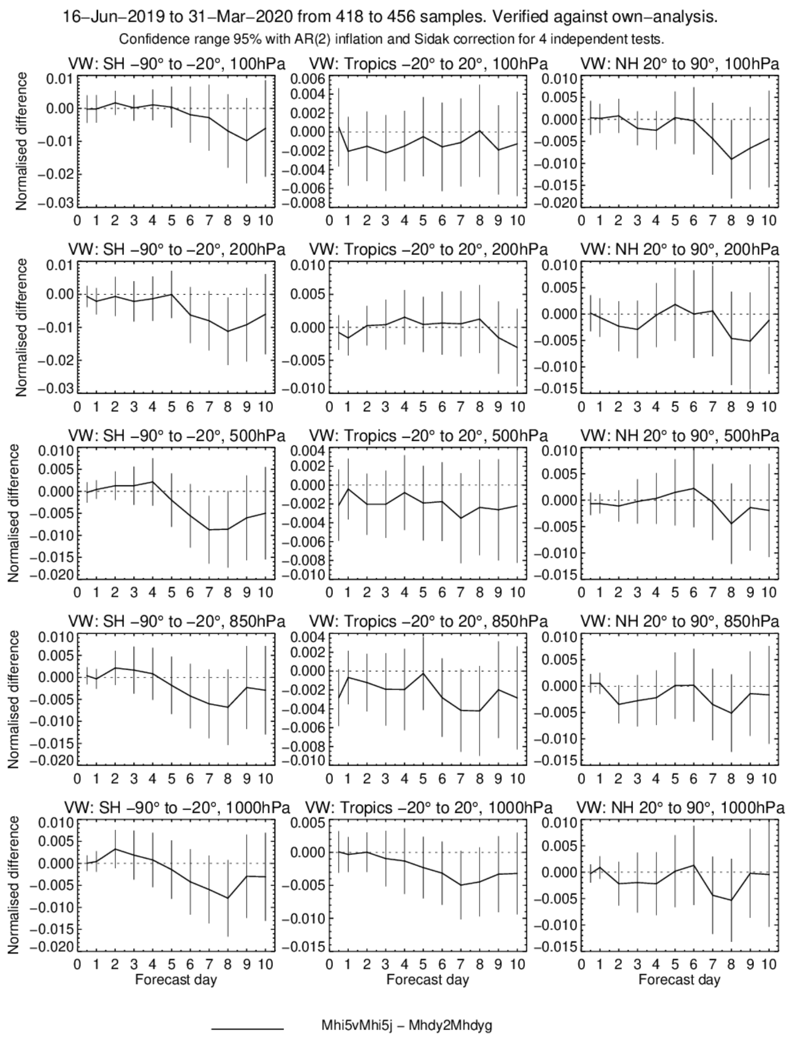
Z-ACC: above 0 better

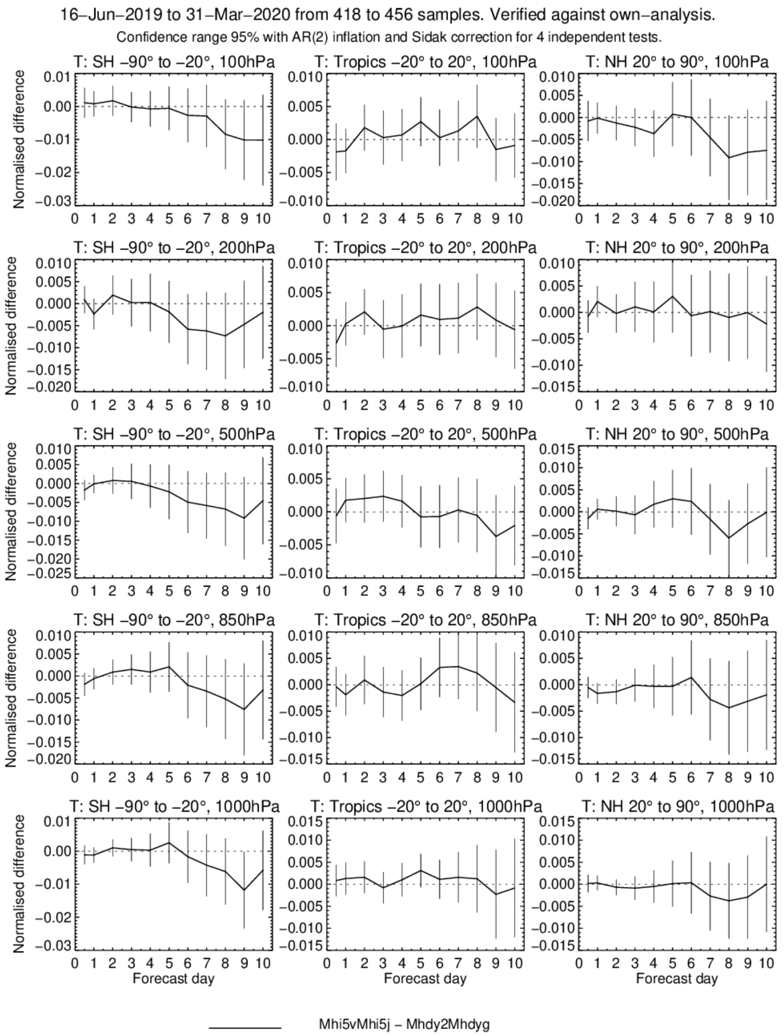
VW-RMSE: below 0 better

T-RMSE: below 0 better

# Combined winter/summer Tco399 4DVAR forecast scores (8 months)




Z-ACC: above 0 better

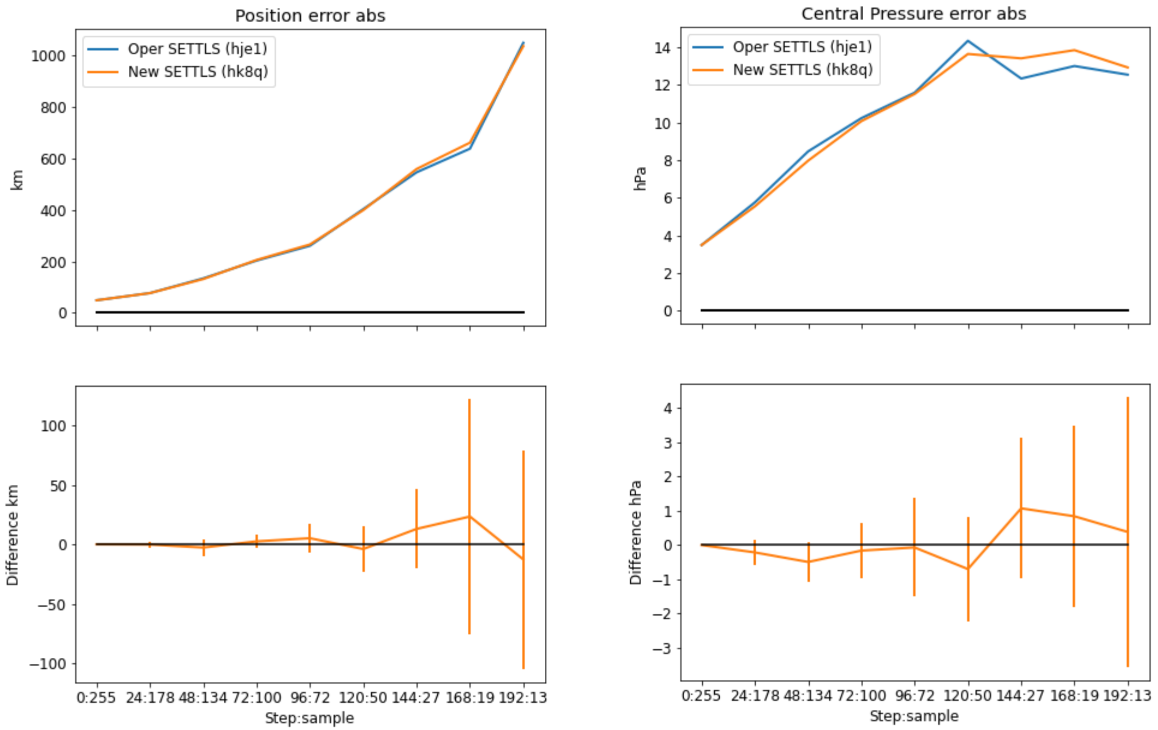
VW-RMSE: below 0 better

T-RMSE: below 0 better

# Tropical cyclones and computational performance in single precision (SP)

- Since May 2021 IFS runs forecasts in SP
- Model is **faster** but **forecast equally accurate**

| Cost for computing DP (9km forecast) | |
|---|---|
| Current operational scheme in SP | 220 ms/step 6.9% of total time-step cost |
| New scheme in SP | 135 ms/step 4.1% of total time-step cost |



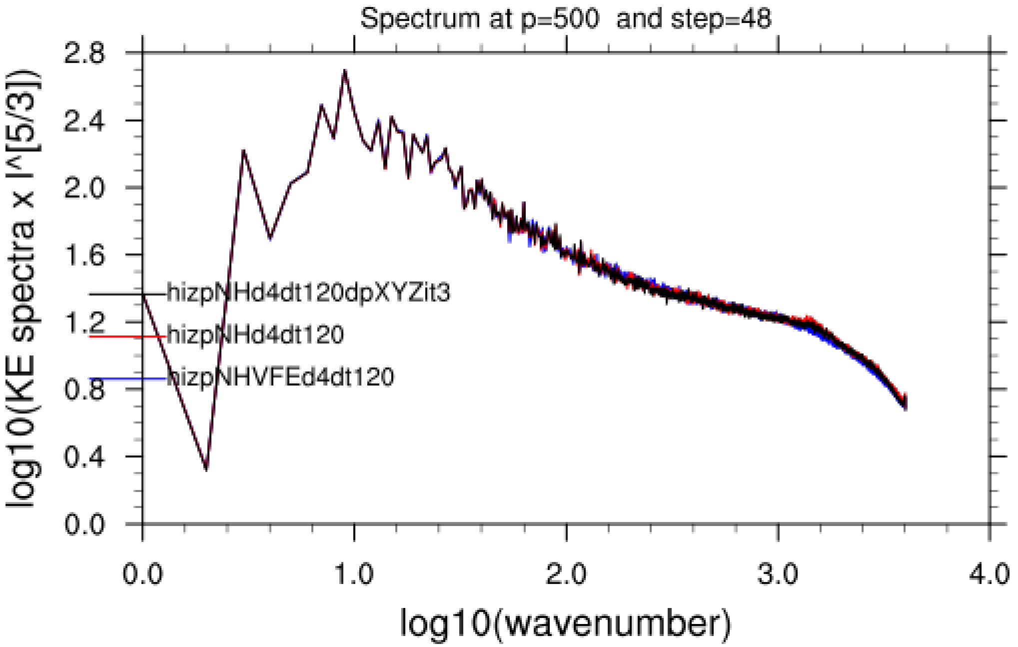

Thanks to Linus Magnusson

North Hemisphere tropical cyclone 2020 season (9km resolution verification against Best Track obs)

**New scheme in SP versus operational in double-precision: 2 x fast model but same accuracy! ☺**

# Global non-hydrostatic testing at 2.5km resolution

- IFS has non-hydrostatic version (member states a key contributor)
- Global IFS NH dynamics are more than 2 x expensive:
  - More equations and variables and spectral transforms (very expensive at high resolution)
  - ICI (predictor-corrector) scheme for stability: **call dynamics twice / step**
- The new SETTLS has been tested successfully in the NH with ICI scheme: the predictor-corrector approach is exploited to converge it fast




500 hPa KE spectra from 3 NH versions at 2.5km res with 137 lev, 120s timestep:

- RED vertical finite difference discretization oper version of SETTLS
- BLUE new vertical finite element (*Vivoda, Smolikova, Simarro MWR 2018)*
- BLACK = BLUE + new SETTLS

# Summary

- A Cartesian implementation of SETTLS is proposed for 48r1 for its efficiency, compactness, advantages for single precision and near pole calculations
- It includes a new starting procedure for the departure point iterative scheme with faster convergence
- Faster cleaner code with same accuracy

**Future work:**

I hope that this work may stimulate future efforts to improve SL parallel communication and development of STOCHDP (stochastic advection scheme led by S-J Lock)

Work currently under review in QJRMS: "*A fast converging and concise algorithm for computing the departure points in semi-Lagrangian weather and climate models*"

**Thank you for your attention!**