Assimilating active and passive microwave observations of precipitation from space

into cloud-permitting models

Ziad S. Haddad + Jeffrey Steward + Tomislava Vukicevic + Svetla Hristova-Veleva
Jet Propulsion Laboratory, California Institute of Technology

- Data assimilation has to become The retrieval algorithm for rain radar, passive microwave

- Effective assimilation of "rainy radiances" still elusive, partly because the model does not represent the rain realistically, and …

- … partly because the observations depend non-linearly on variables and parameters whose values and evolution are not well understood

⇒ try to improve the representation of obs

Challenge is to find $\vec{p}$ such that

$$(\vec{p}-\vec{p}_b)^t\, B^{-1}\,(\vec{p}-\vec{p}_b)\; +\; \left(\vec{Z}-\mathcal{Z}(\vec{p})\right)^t R^{-1}\left(\vec{Z}-\mathcal{Z}(\vec{p})\right)$$

is minimized. In the case of radiometer observations, need to minimize

$$(\vec{p}-\vec{p}_b)^t\, B^{-1}\,(\vec{p}-\vec{p}_b)\; +\; \left(\vec{T}_b-\mathcal{RT}(\vec{p})\right)^t R^{-1}\left(\vec{T}_b-\mathcal{RT}(\vec{p})\right)$$

$$T_b \;=\; \underset{\uparrow}{\mathcal{RT}}\,(p_1,\ldots,p_{504};\lambda_1,\underset{\uparrow}{\lambda_2},\ldots)$$

very nonlinear

oversimplified parameters

Challenge is to find $\vec{p}$ such that

$$(\vec{p}-\vec{p}_b)^t\, B^{-1}\,(\vec{p}-\vec{p}_b) \;+\; \left(\vec{Z}-\mathcal{Z}(\vec{p})\right)^t R^{-1}\left(\vec{Z}-\mathcal{Z}(\vec{p})\right)$$

is minimized. In the case of radiometer observations, need to minimize

$$(\vec{p}-\vec{p}_b)^t\, B^{-1}\,(\vec{p}-\vec{p}_b) \;+\; \left(\vec{T}_b-\mathcal{RT}(\vec{p})\right)^t R^{-1}\left(\vec{T}_b-\mathcal{RT}(\vec{p})\right)$$

Currently:

$$T_b \;=\; \mathcal{RT}\left(p_1, ..., p_{504}; \lambda_1, \lambda_2, ...\right)$$

very nonlinear

each of these is
a **single value**

each of these represents a
volume-element **mean**

Challenge is to find $\vec{p}$ such that

$$(\vec{p} - \vec{p}_b)^t\, B^{-1}\, (\vec{p} - \vec{p}_b) \;+\; \left(\vec{Z} - \mathcal{Z}(\vec{p})\right)^t R^{-1} \left(\vec{Z} - \mathcal{Z}(\vec{p})\right)$$

is minimized. In the case of radiometer observations, need to minimize

$$(\vec{p} - \vec{p}_b)^t\, B^{-1}\, (\vec{p} - \vec{p}_b) \;+\; \left(\vec{T}_b - \mathcal{RT}(\vec{p})\right)^t R^{-1} \left(\vec{T}_b - \mathcal{RT}(\vec{p})\right)$$

Currently:

$$T_b \;=\; \mathcal{RT}\,(p_1, ..., p_{504}; \lambda_1, \lambda_2, ...)$$

$\Rightarrow$ cannot reflect variable distribution within volume element

$\Rightarrow$ **cannot account for “uncertainty” in the parameter values**

How bad is the non-linearity problem?

$$\text{water content} \;=\; q \;=\; \int_0^\infty \frac{4}{3}\pi\left(\frac{D}{2}\right)^3 \rho \cdot N(D)dD$$

$$\text{observation} \;=\; Z \;=\; \int_0^\infty \sigma_b(D)\;\cdot\;N(D)\,dD$$

where

$N(D)dD \;=\;$ number of drops of diameter between D and $D + dD$

$$Z \;=\; \mathcal{F}(\,q\,,\text{parameters of } N)$$

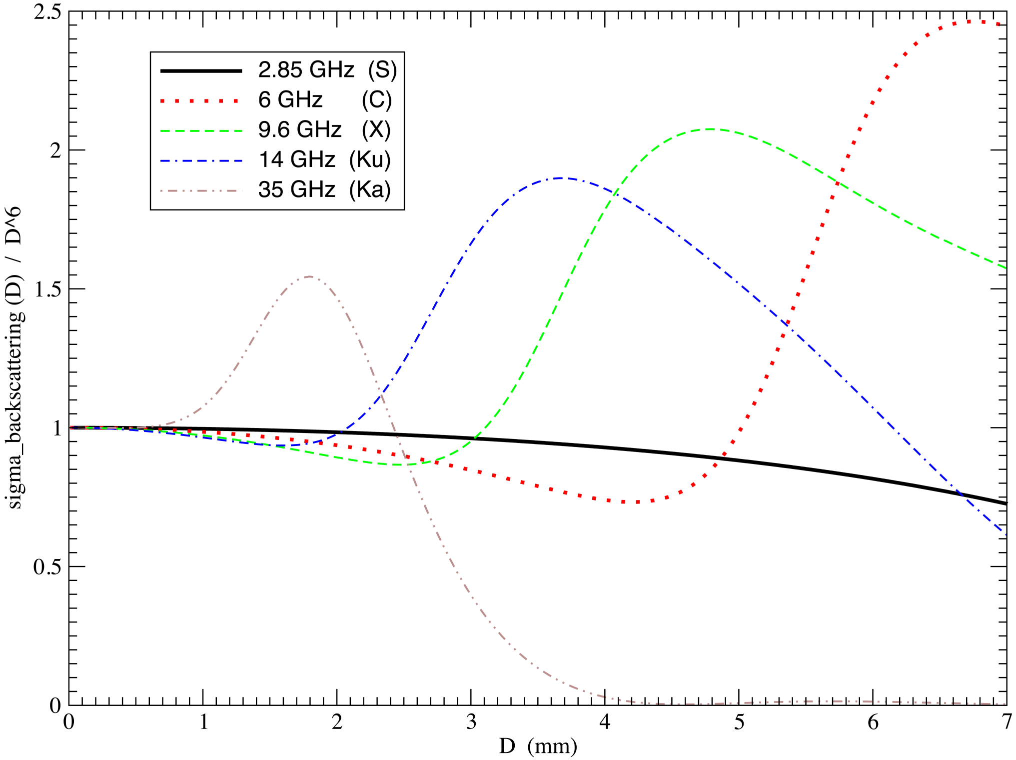
How bad is the non-linearity problem?

$$\text{water content} \; = \; q \; = \; \int_0^\infty \frac{4}{3}\pi \left(\frac{D}{2}\right)^3 \rho \cdot N(D) dD$$

$$\text{observation} \; = \; Z \; = \; \int_0^\infty \sigma_b(D) \cdot N(D)\, dD$$

where

$N(D)dD \; = \;$ number of drops of diameter between D and $D + dD$



How bad is the non-linearity problem?

$$\text{water content} \;=\; q \;=\; \int_0^\infty \frac{4}{3}\pi\left(\frac{D}{2}\right)^3 \rho\cdot N(D)dD$$

$$\text{observation} \;=\; Z \;=\; \int_0^\infty \sigma_b(D)\;\cdot\; N(D)\,dD$$

where

$N(D)dD \;=\;$ number of drops of diameter between D and $D+dD$

For every diameter D_i, define a

"monodisperse distribution" $N(D) \;=\; N_i\delta(D-D_i)$

$$q \;=\; N_1\frac{\pi\rho}{6}D_1^3 \;=\; N_2\frac{\pi\rho}{6}D_2^3$$

$$Z_1 = N_1\sigma_b(D_1) \qquad Z_2 = N_2\sigma_b(D_2)$$

How bad is the non-linearity problem?

$$\text{water content} \;=\; q \;=\; \int_0^\infty \frac{4}{3}\pi\left(\frac{D}{2}\right)^3 \rho\cdot N(D)dD$$

$$\text{observation} \;=\; Z \;=\; \int_0^\infty \sigma_b(D)\;\cdot\;N(D)\,dD$$

where

$N(D)dD \;=\;$ number of drops of diameter between D and $D + dD$

For every diameter $D_i,$ define a

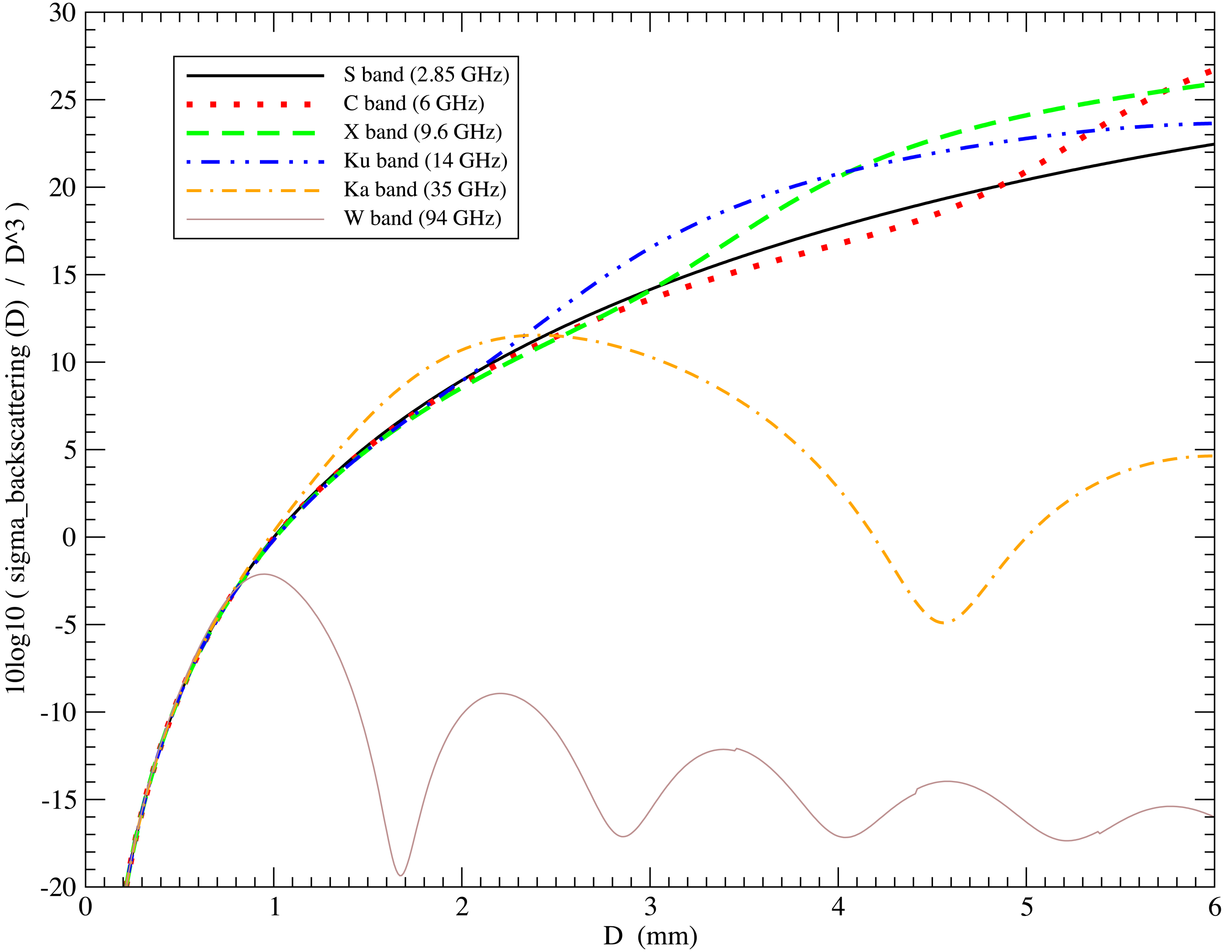
"monodisperse distribution" $N(D) \;=\; N_i\delta(D - D_i)$

$$q \;=\; N_1\frac{\pi\rho}{6}D_1^3 \;=\; N_2\frac{\pi\rho}{6}D_2^3 \quad\Rightarrow\quad N_2/N_1 \;=\; (D_1/D_2)^3$$

$$Z_1 = N_1\sigma_b(D_1) \qquad Z_2 = N_2\sigma_b(D_2)$$

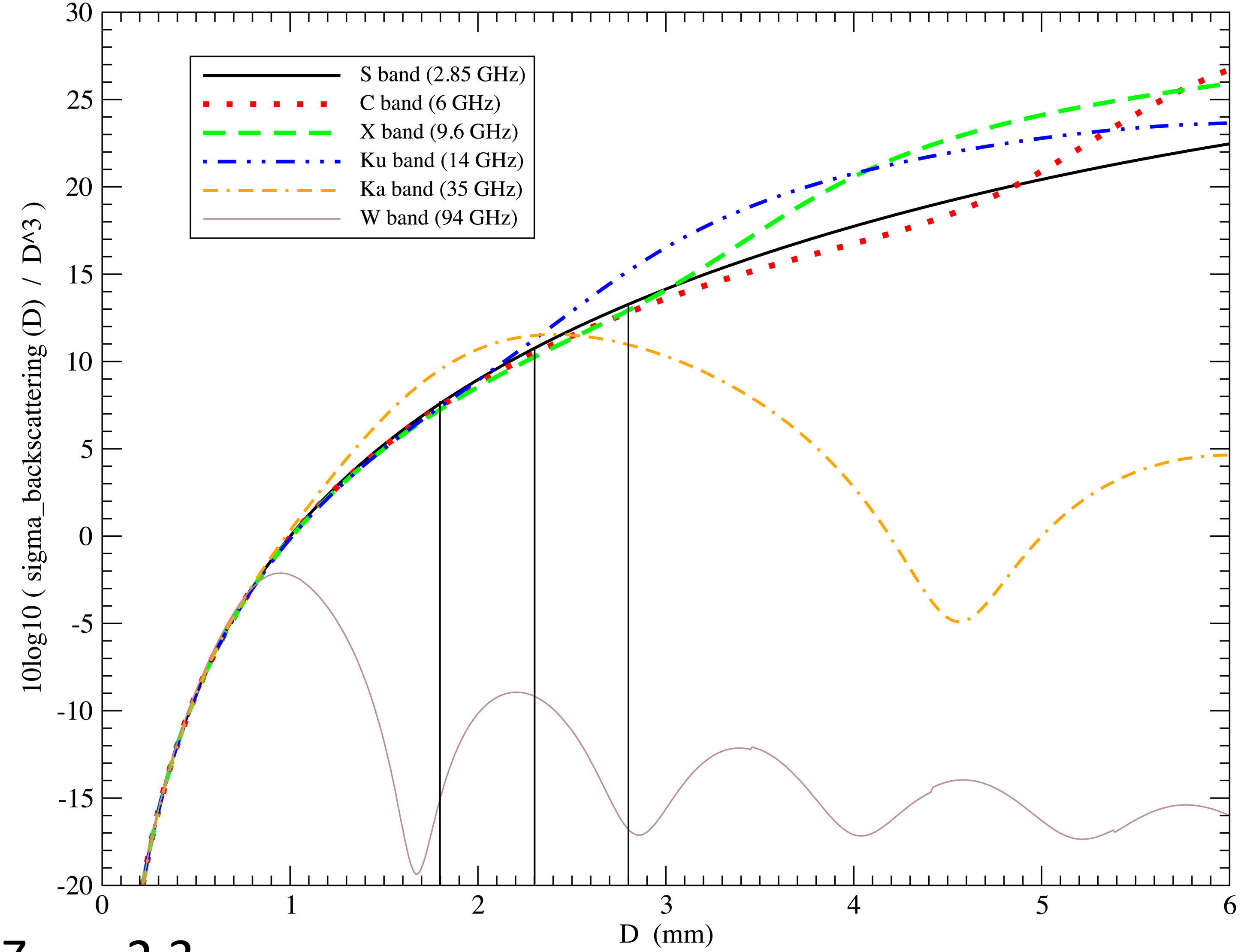
$$\Rightarrow\quad Z_2/Z_1 \;=\; \frac{\sigma_b(D_2)/D_2^3}{\sigma_b(D_1)/D_1^3}$$

How bad is the non-linearity problem?



$$\Rightarrow \quad Z_2/Z_1 = \frac{\sigma_b(D_2)/D_2^3}{\sigma_b(D_1)/D_1^3}$$

How bad is the non-linearity problem?



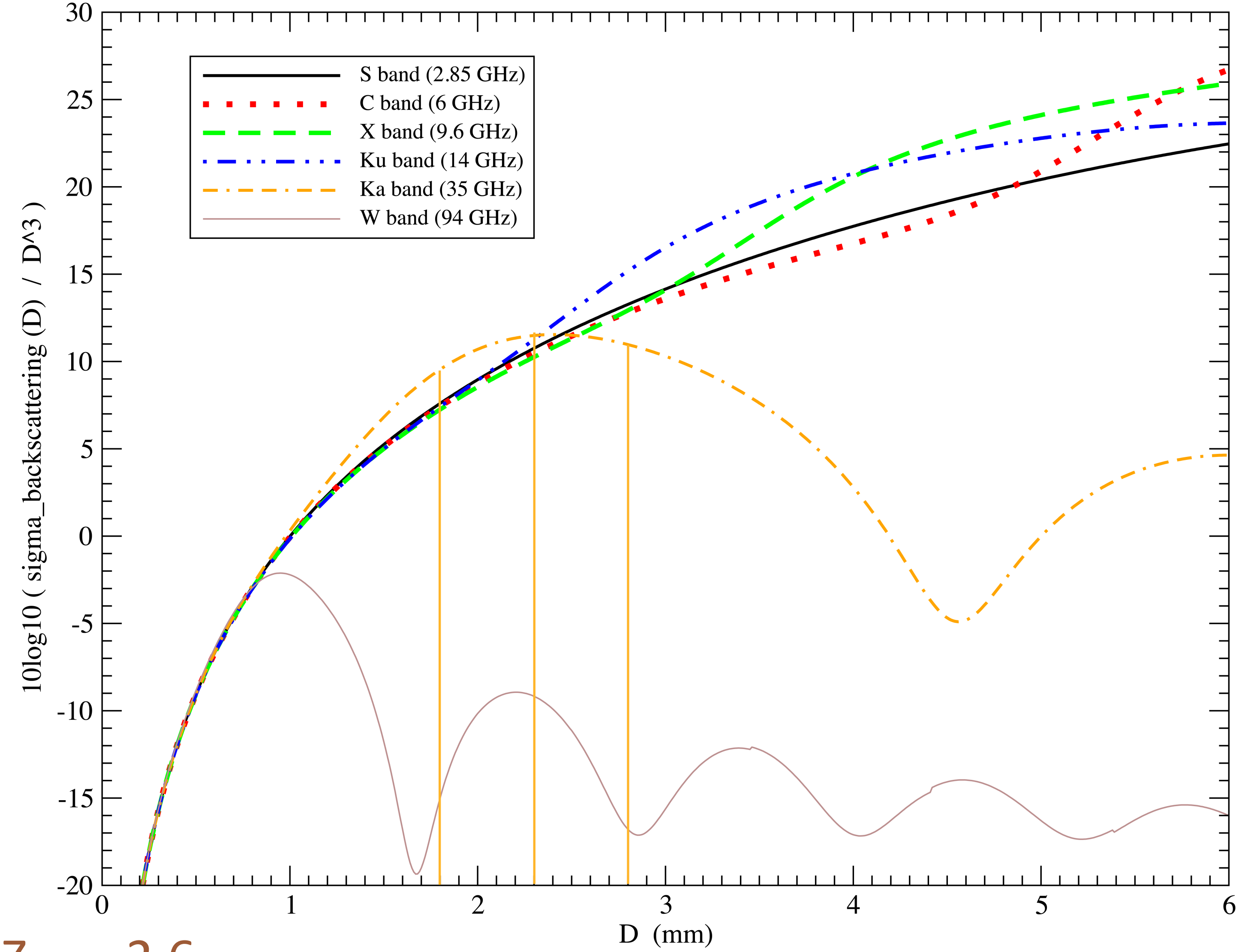
S band:

$dbZ_{2.3}$ - $dbZ_{1.8}$ = 2.3

$dbZ_{2.8}$ - $dbZ_{2.3}$ = 2.6

$$\Rightarrow \quad Z_2/Z_1 \;=\; \frac{\sigma_b(D_2)/D_2^3}{\sigma_b(D_1)/D_1^3}$$

How bad is the non-linearity problem?



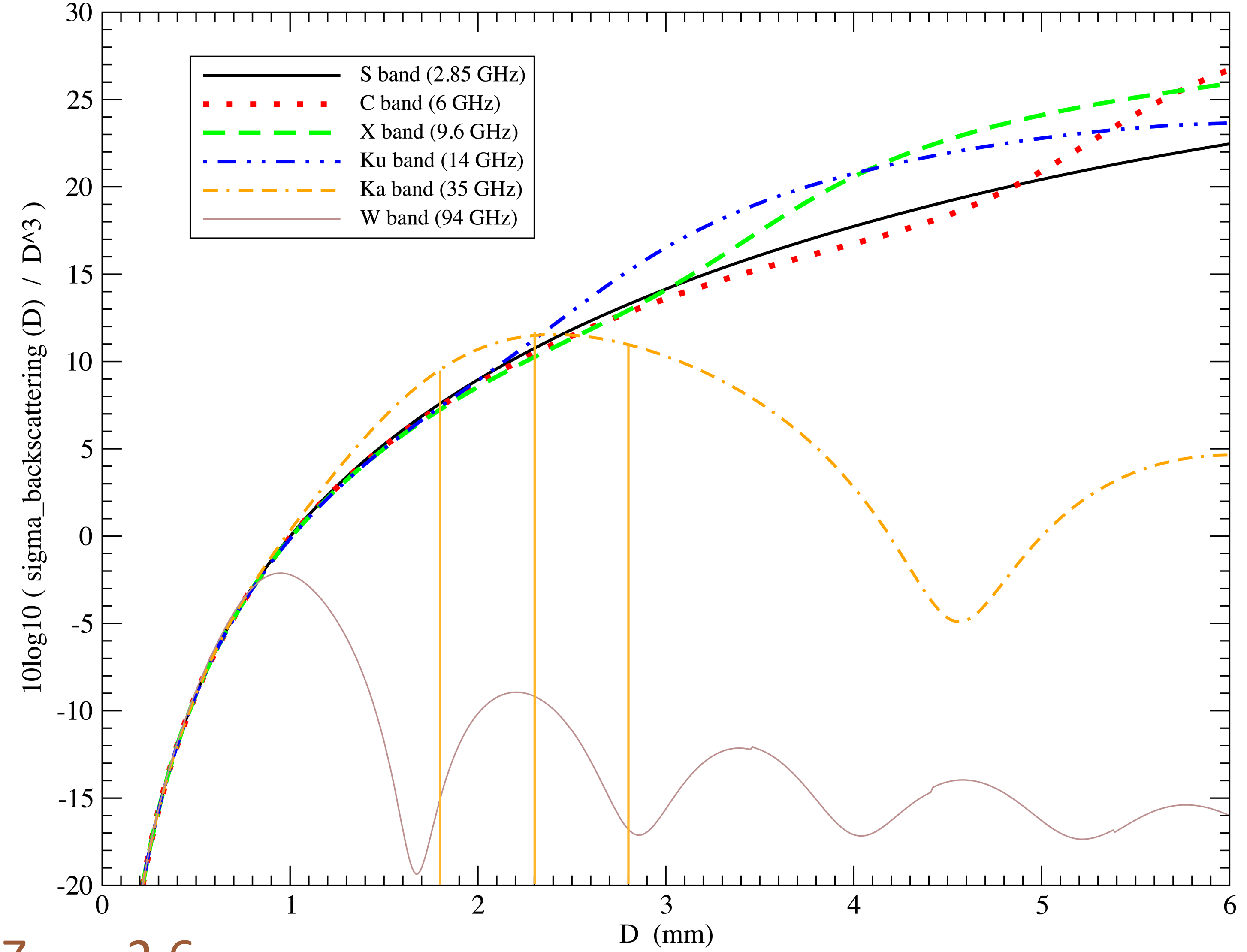
Ka band:

$dbZ_{2.3}$ - $dbZ_{1.8}$ = 2.6

$dbZ_{2.8}$ - $dbZ_{2.3}$ = -0.9

$$\Rightarrow \quad Z_2/Z_1 \; = \; \frac{\sigma_b(D_2)/D_2^3}{\sigma_b(D_1)/D_1^3}$$

How bad is the non-linearity problem?



Ka band:

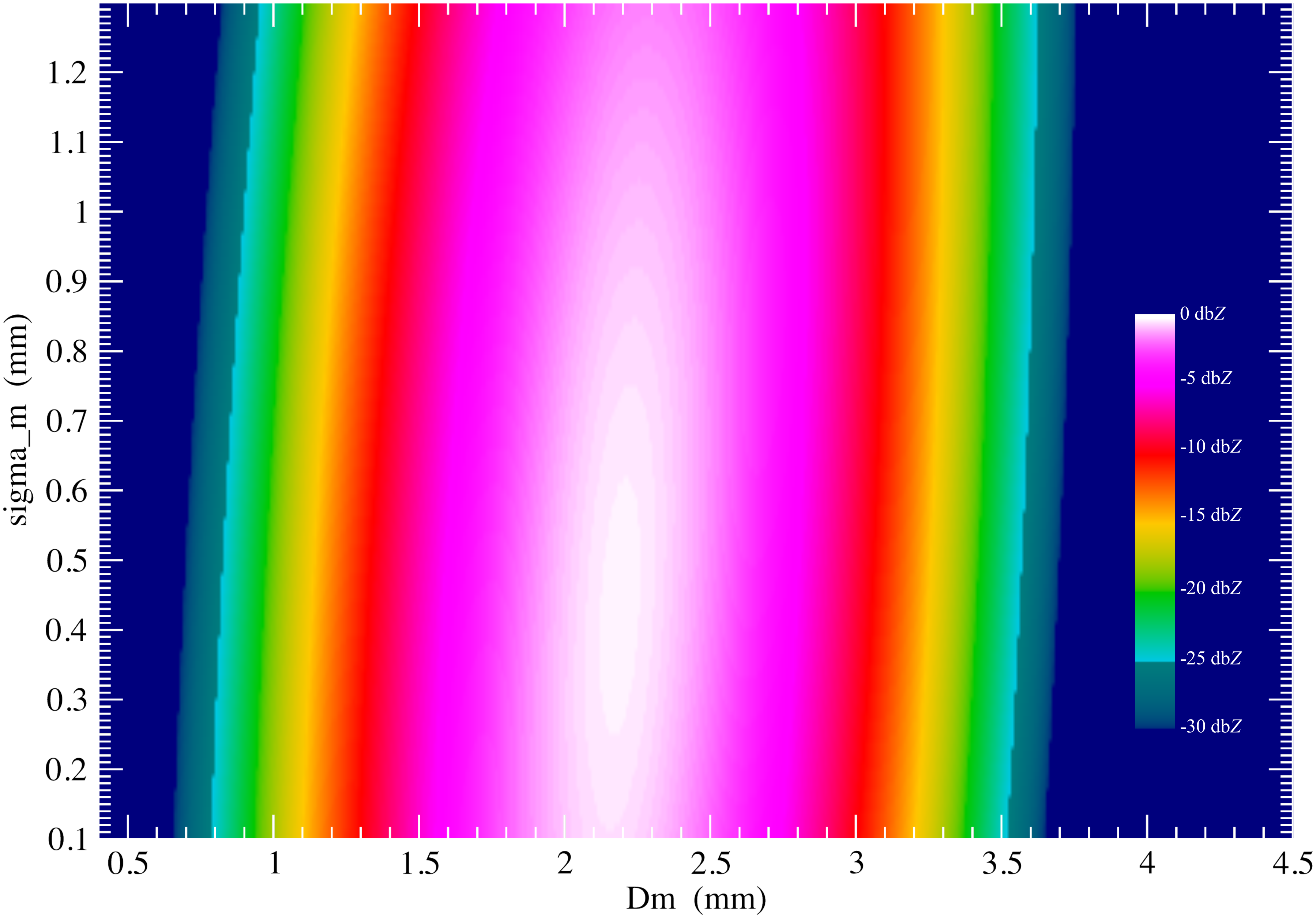
$dbZ_{2.3}$ - $dbZ_{1.8}$ = 2.6

$dbZ_{2.8}$ - $dbZ_{2.3}$ = -0.9

$$Z = \mathcal{F}(q \text{ , parameters of } N)$$

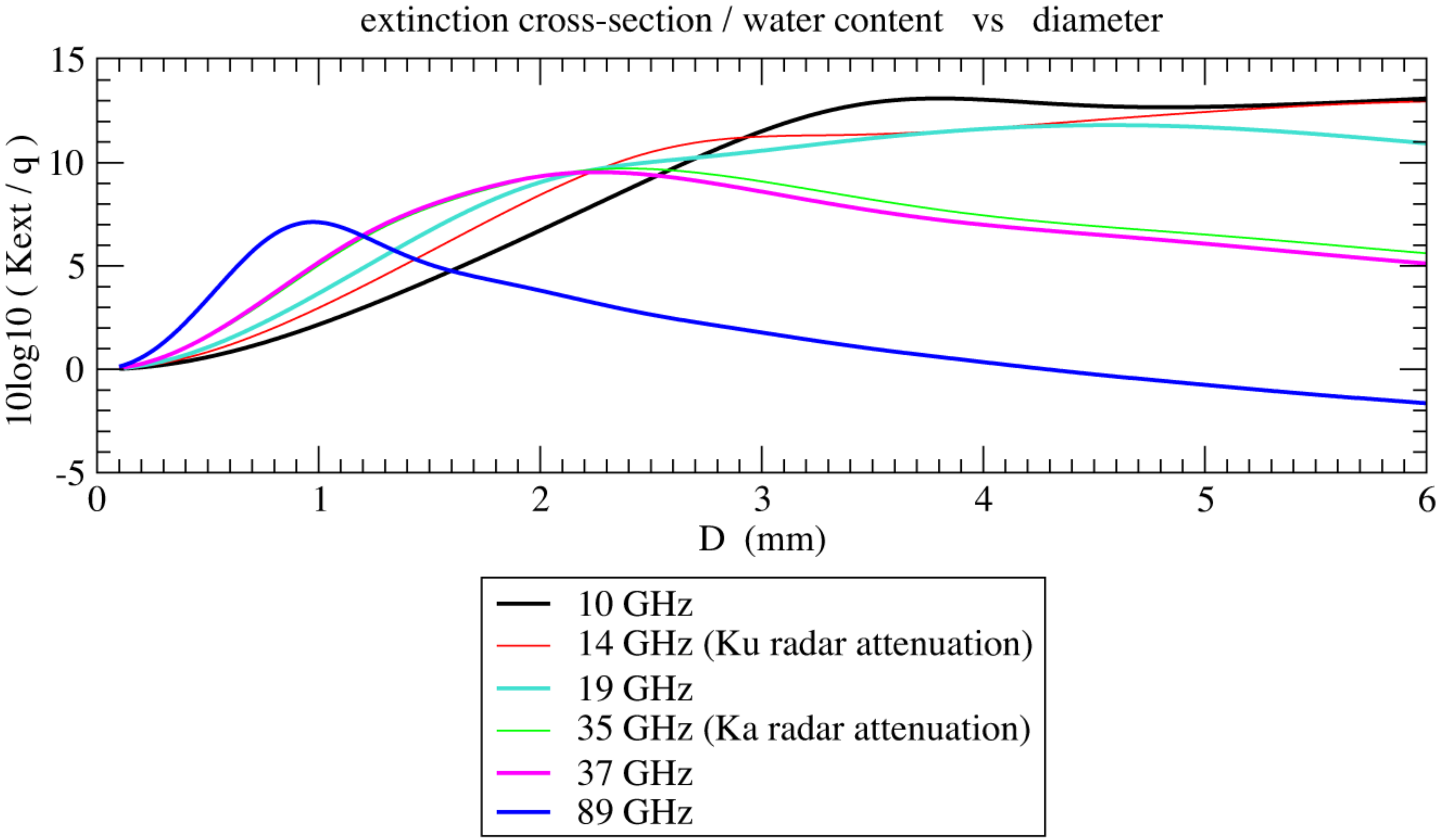
$\Rightarrow$ "parameter D = mean (2)" will not produce the mean Z

$\mathrm{db}Z(q;\, D_m,\, \sigma_m) \;-\; \mathrm{db}Z(q;\, 2.3,\, 0.4)$



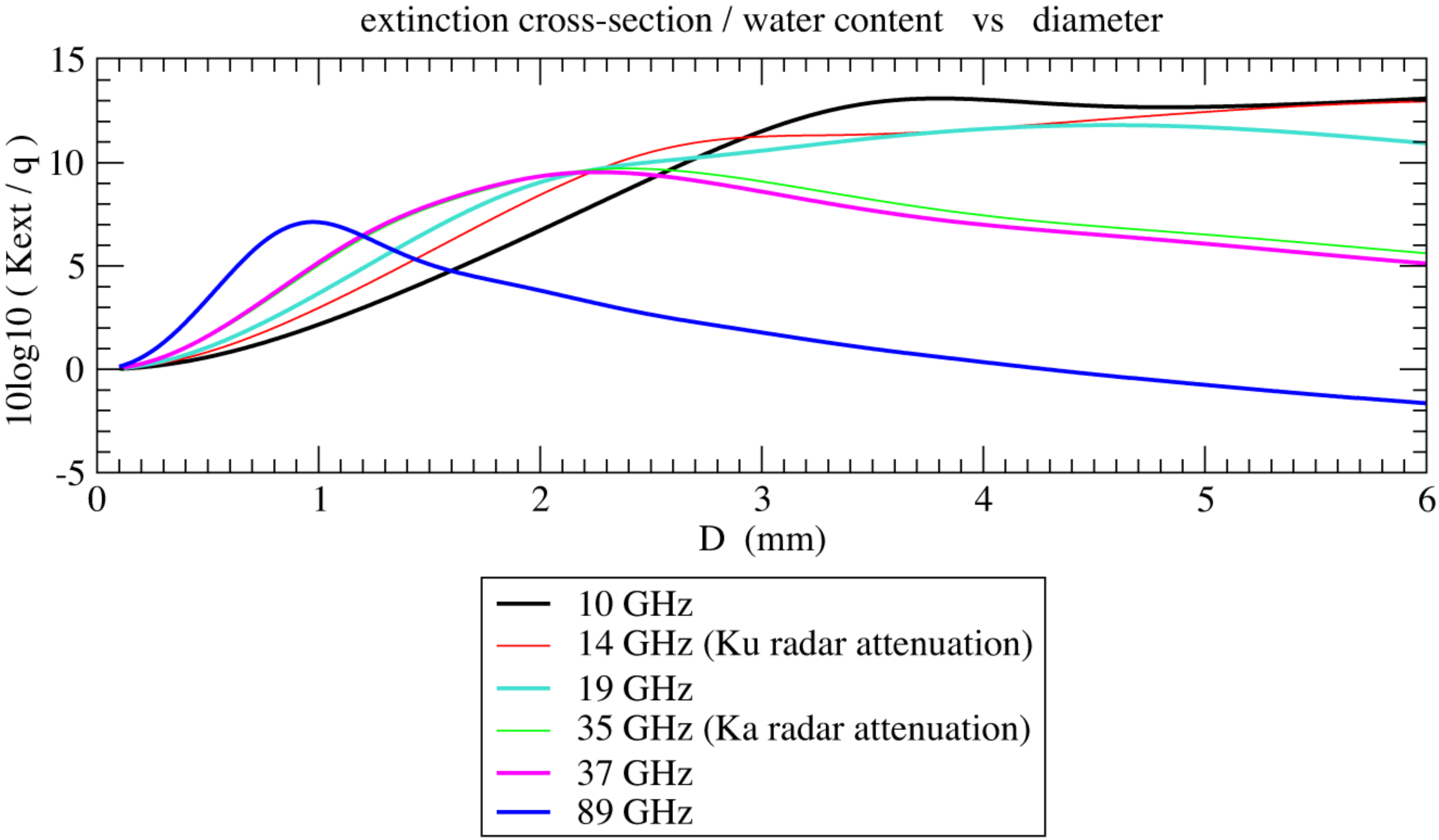
How bad is the non-linearity problem?

In the passive case:



How bad is the non-linearity problem?

In the passive case:



and don’t forget: **“monodisperse” assumption is not realistic!**

Challenge is to find $\vec{p}$ such that

$$(\vec{p}-\vec{p}_b)^t\, B^{-1}\,(\vec{p}-\vec{p}_b)\; +\; \left(\vec{Z}-\mathcal{Z}(\vec{p})\right)^t R^{-1}\left(\vec{Z}-\mathcal{Z}(\vec{p})\right)$$

is minimized. In the case of radiometer observations, need to minimize

$$(\vec{p}-\vec{p}_b)^t\, B^{-1}\,(\vec{p}-\vec{p}_b)\; +\; \left(\vec{T}_b-\mathcal{RT}(\vec{p})\right)^t R^{-1}\left(\vec{T}_b-\mathcal{RT}(\vec{p})\right)$$

Currently:

$$T_b \;=\; \mathcal{RT}\,(p_1,...,p_{504};\lambda_1,\lambda_2,...)$$

$\Rightarrow$ cannot reflect variable distribution within volume element

$\Rightarrow$ **cannot account for "uncertainty" in the parameter values**

$\Rightarrow$ look for a way to calculate $\mathcal{RT}$ that does not just use 1 deterministic forward calculation using means as input

Instead: look for an "empirical representation" of T_b

⇒ so:

1. off-line, for a given (x_1, x_2, …, x_{504}), calculate radiances with different λ, and perhaps different sub-resolution distributions
2. store the answers in a large database
3. in real-time, for a given (x_1, x_2, …, x_{504}), calculate radiances by referring to the database:

$$T_b\,(x_1, x_2, \ldots, x_{504}) \;=\; \frac{\Sigma \;\; T_b^{(n)} \exp(-[x_1-x_1^{(n)}]^2 -[x_2-x_2^{(n)}]^2 - \ldots -[x_{504}-x_{504}^{(n)}]^2)}{\Sigma \;\; \exp(-[x_1-x_1^{(n)}]^2 -[x_2-x_2^{(n)}]^2 - \ldots -[x_{504}-x_{504}^{(n)}]^2)}$$

Obvious problem with this:

- Too many variables (cannot work in 504-dimensional space),
- Not all of these variables are important for T_b

Note: T_b calculated exactly (off-line) using forward radiative transfer

Instead: look for an "empirical representation" of T_b

⇒ so:

1. off-line, for a given (x_1, x_2, …, x_{504}), calculate radiances with different λ, **different sub-resolution distributions, different RT**
2. store the answers in a large database
3. in real-time, for a given (x_1, x_2, …, x_{504}), calculate radiances by referring to the database:

$$T_b(x_1, x_2, \ldots, x_{504}) = \frac{\Sigma\ T_b^{(n)} \exp(-[x_1-x_1^{(n)}]^2 - [x_2-x_2^{(n)}]^2 - \ldots - [x_{504}-x_{504}^{(n)}]^2)}{\Sigma\ \exp(-[x_1-x_1^{(n)}]^2 - [x_2-x_2^{(n)}]^2 - \ldots - [x_{504}-x_{504}^{(n)}]^2)}$$

⇒ so: "distill" the variables (x_1, x_2, …, x_{504}) into a smaller number which would capture the main sensitivities of the brightness temperatures, namely

- the absorption/emission and the scattering
- the surface temperature and the wind speed

so about 4 distilled variables y_1, y_2, y_3, y_4 for the entire column:

$$T'(x_1, x_2, \ldots, x_{504}) = \frac{\Sigma\ T'^{(n)} \exp(-[y_1-y_1^{(n)}]^2-[y_2-y_2^{(n)}]^2-[y_3-y_3^{(n)}]^2-[y_4-y_4^{(n)}]^2)}{\Sigma\ \exp(-[y_1-y_1^{(n)}]^2-[y_2-y_2^{(n)}]^2-[y_3-y_3^{(n)}]^2-[y_4-y_4^{(n)}]^2)}$$

Our methodology to derive our operator:

- Start with HWRF simulations (say HEDAS Earl 2010 h3vk, 2010-08-29-12Z to 2010-09-03-18Z), using stream ψ, potential χ, P, T, RH, W, q_{cliq}, q_r, q_{cli}, q_s, q_g, q_h at 42 vertical levels for a total of 504 variables $x_1, \ldots , x_{504}$
- for each of these 12million columns, forward-calculate $T_{b1} , \ldots , T_{b9}$

- find the principal components $x_1', \ldots , x_{504}'$ (each is a linear combo of $x_1, \ldots , x_{504}$) and the principal components $T_1', \ldots , T_9'$ (each a linear combo of $T_1' , \ldots , T_9'$)

- Then we will have to find combos of $x_1',\ldots, x_{504}'$ that correlate most with combos of $T_1',\ldots, T_9'$

- Say these combos are x_1'', x_2'', x_3'' and T_1'', T_2'', T_3'': we finally need to express the latter in terms of the former, in a differentiable way (to be able to compute derivatives)

Our methodology to derive our operator:

- Start with HWRF simulations (say HEDAS Earl 2010 h3vk, 2010-08-29-12Z to 2010-09-03-18Z), using stream ψ, potential χ, P, T, RH, W, q_{cliq}, q_r, q_{cli}, q_s, q_g, q_h at 42 vertical levels for a total of 504 variables x_1, … , x_{504}
- for each of these 12million columns, forward-calculate T_{b1} , … , T_{b9}

- Step 1: find the principal components x_1', … , x_{504}'
- Step 2: find the principal components T_1', … , T_9'

- Step 3: find
  combos of $\underset{w_T}{x_1',\dots,}\, x_{504}'$ that correlate most with combos of $\underset{w_x}{T_1',\dots,}\, T_9'$

$$\left(\mathrm{Cov}_T^{-1}\cdot\mathrm{Cov}_{T,x}\right)\;\cdot\;\left(\mathrm{Cov}_x^{-1}\cdot\mathrm{Cov}_{T,x}^{t}\right)\;w_T\;=\;\lambda\,w_T$$

$$\left(\mathrm{Cov}_x^{-1}\cdot\mathrm{Cov}_{T,x}^{t}\right)\;\cdot\;\left(\mathrm{Cov}_T^{-1}\cdot\mathrm{Cov}_{T,x}\right)\;w_x\;=\;\lambda\,w_x$$

Our methodology to derive our operator:

- Start with HWRF simulations (say HEDAS Earl 2010 h3vk, 2010-08-29-12Z to 2010-09-03-18Z), using stream ψ, potential χ, P, T, RH, W, q_{cliq}, q_r, q_{cli}, q_s, q_g, q_h at 42 vertical levels for a total of 504 variables x_1, … , x_{504}
- for each of these 12million columns, forward-calculate T_{b1} , … , T_{b9}

- Step 1: find the principal components x_1', … , x_{504}'
- Step 2: find the principal components T_1', … , T_9'

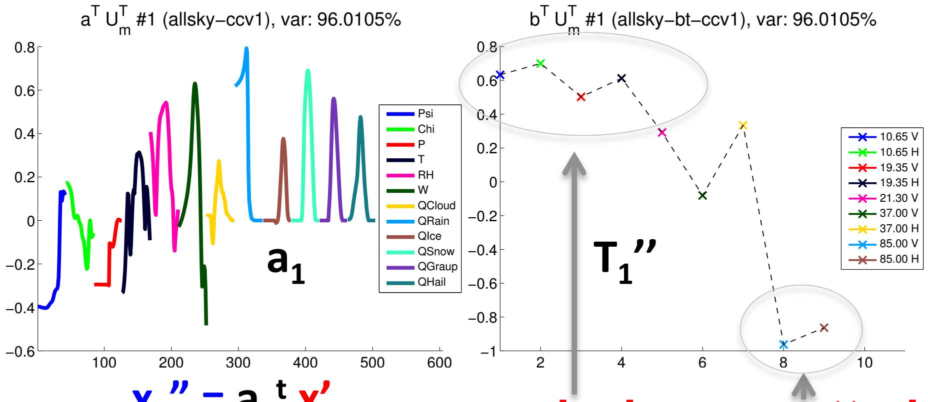
- Step 3: find combos of x_1',…, x_{504}' that correlate most with combos of T_1',…, T_9'

  and express T_1'', T_2'', T_3'' in terms of x_1'', x_2'', x_3'' (with differentiable expression, in order to compute derivatives):

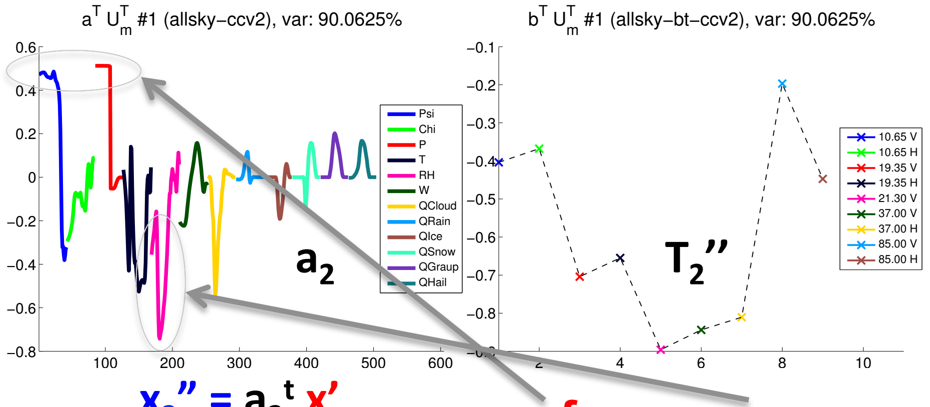
$$T_i''(x_1'', x_2'', x_3'') = \sum T_i''^{(n)} \exp(-[x_1''-x_1''^{(n)}]^2 -[x_2''-x_2''^{(n)}]^2 -[x_3''-x_3''^{(n)}]^2)$$

where the weighted sum over **n runs over the 100million** training "points" (columns)

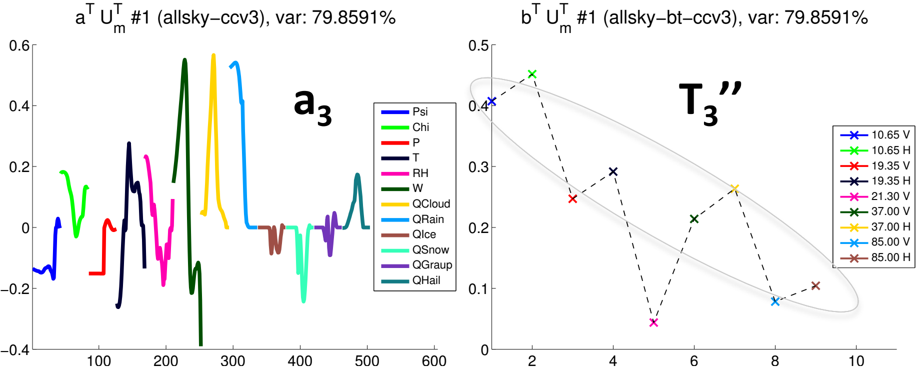
First part of step 3: here are the first 3 x” and T”



$x_1'' = a_1^t\ x'$ **emission – scattering**



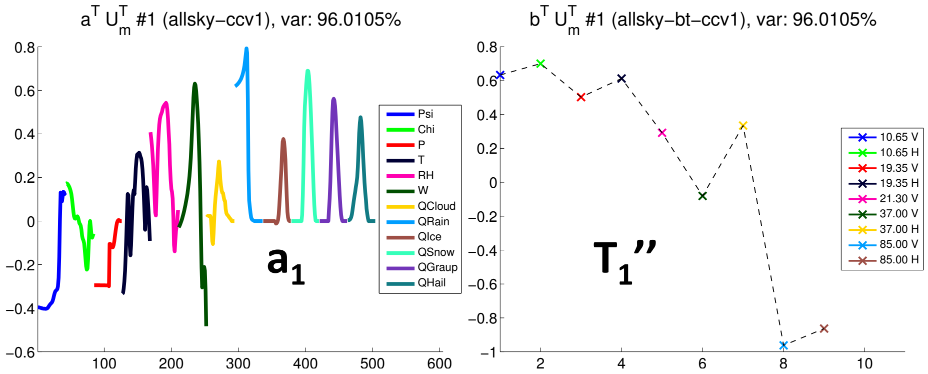
$x_2'' = a_2^t\ x'$ **surface – vapor**



$x_3'' = a_3^t\ x'$ $T_3'' = b_3^t\ T'$

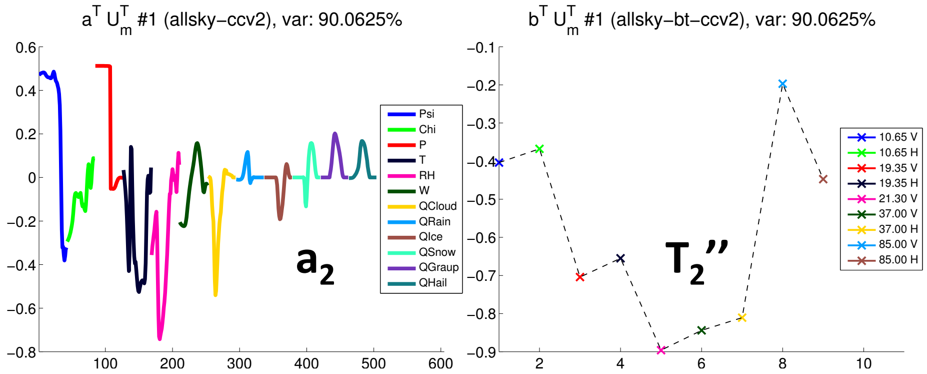
rain (and +correlates) – snow

First part of step 3: here are the first 3 x” and T”



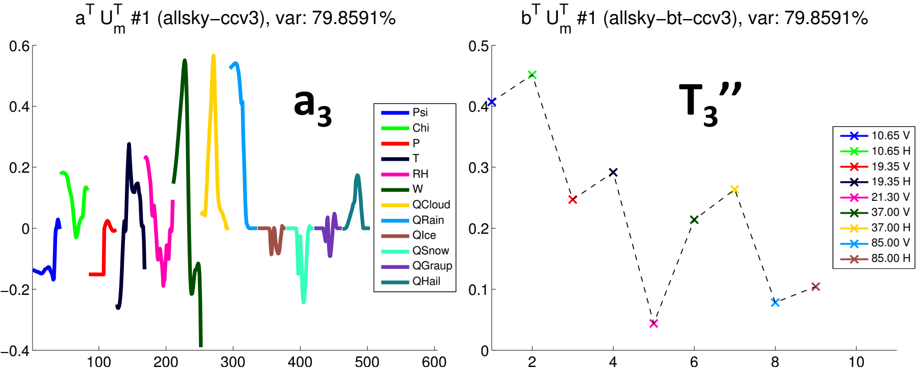
$x_1'' = a_1^t x'$

emission – scattering



$x_2'' = a_2^t x'$

surface – vapor



$x_3'' = a_3^t x'$ **$T_3'' = b_3^t T'$**

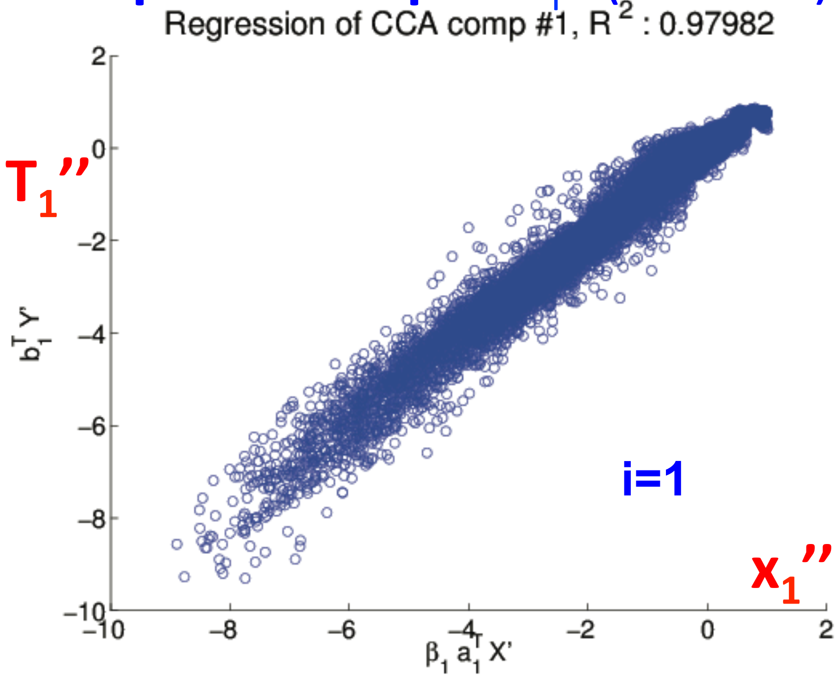
rain (and +correlates) – snow

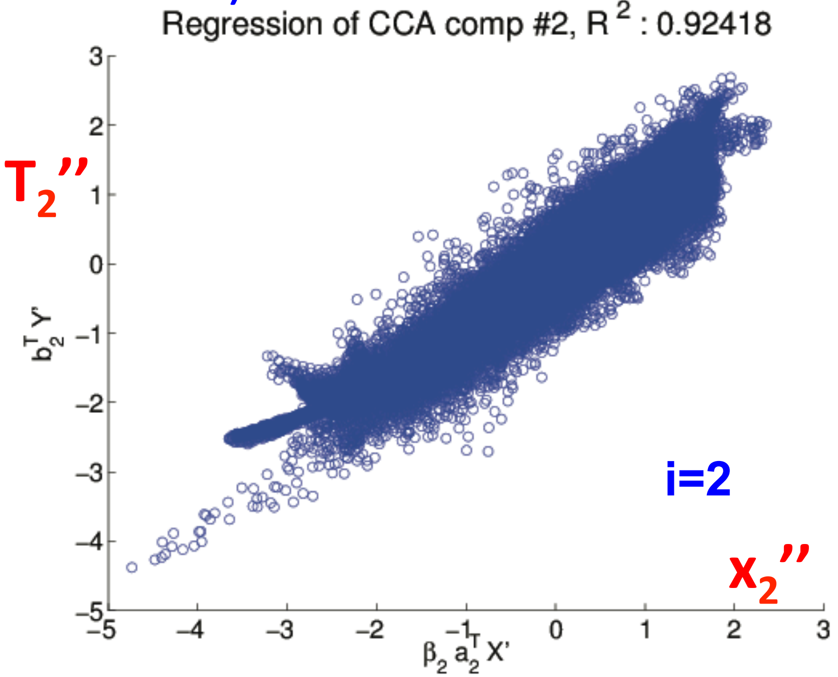
Most remarkable:
the operators H_1, H_2 , H_3
giving

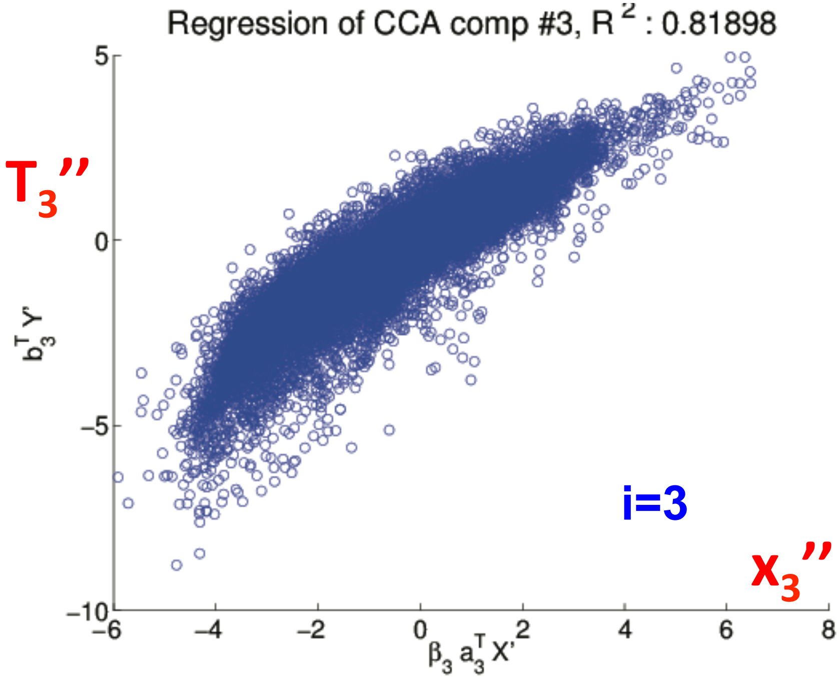
$$T_1'' = H_1(x_1'', x_2'', x_3'')$$
$$T_2'' = H_2(x_1'', x_2'', x_3'')$$
$$T_3'' = H_3(x_1'', x_2'', x_3'')$$

are not so nonlinear:

First part of step 3: T_i'' (vertical) vs x_i'' (horizontal)



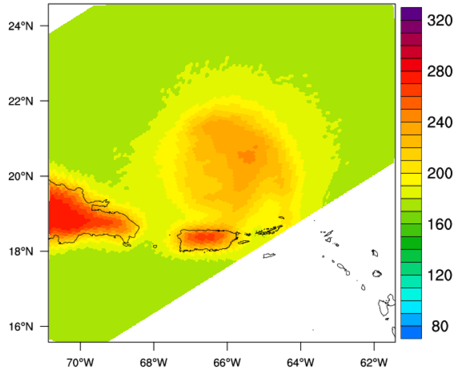


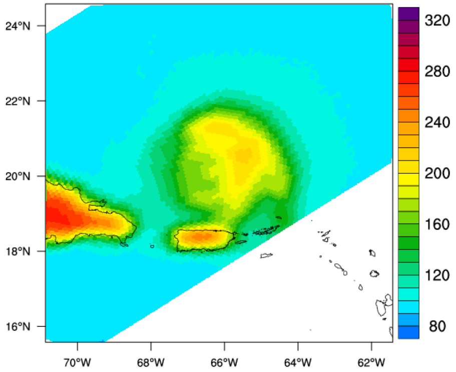


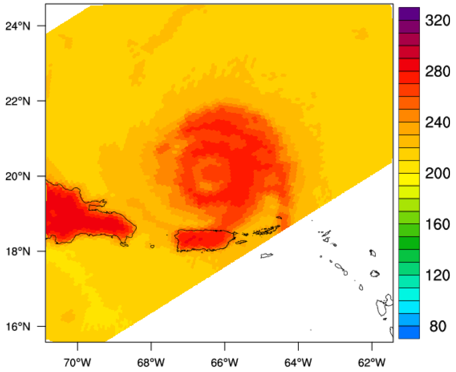
derived observations
vs
derived variables

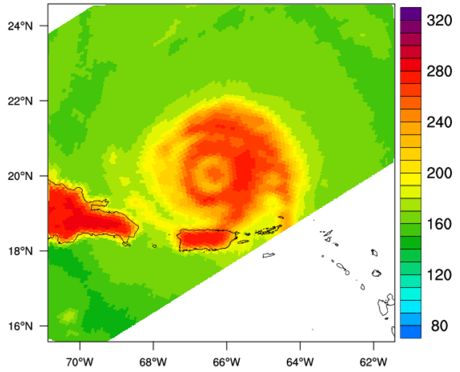
as seen in GRIP's Earl 2010

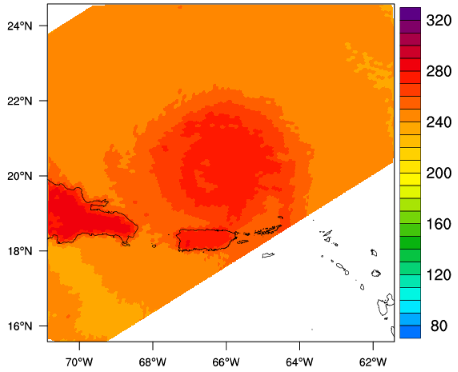
Step 4: EnKF assimilate - GRIP experiment (try #1, keep model simple)

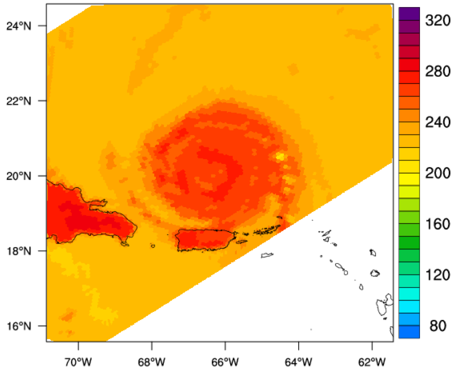

TB for TRMM ch 1 (10.65 GHz V)

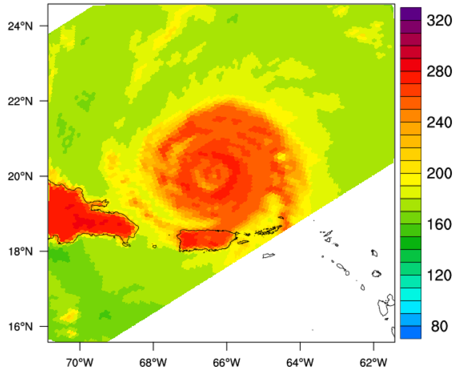

TB for TRMM ch 2 (10.65 GHz H)

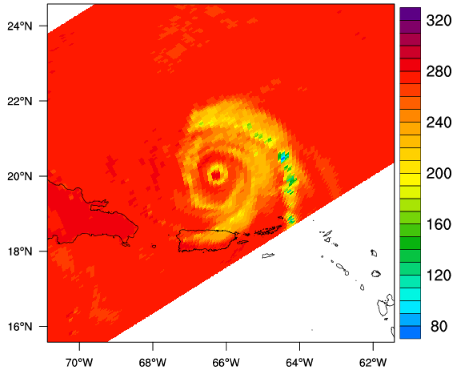

TB for TRMM ch 3 (19.35 GHz V)

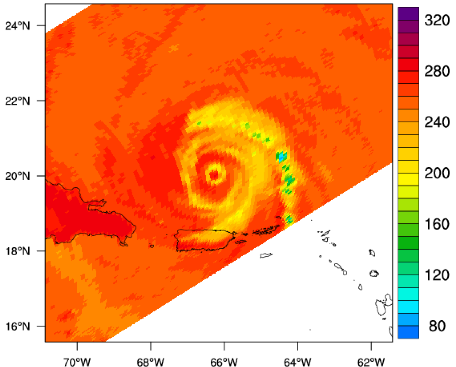

TB for TRMM ch 4 (19.35 GHz H)


TB for TRMM ch 5 (21.30 GHz V)

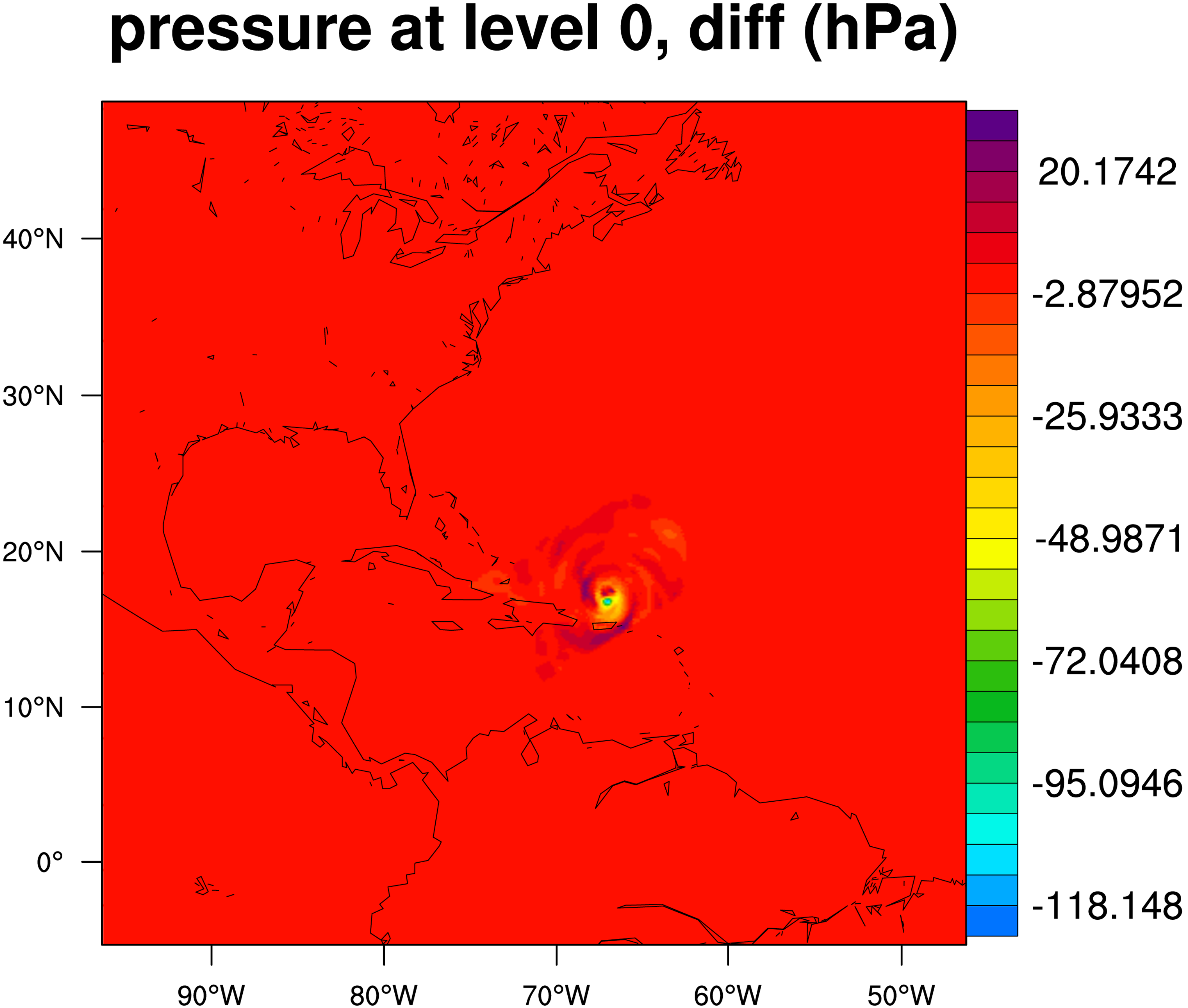

TB for TRMM ch 6 (37.00 GHz V)


TB for TRMM ch 7 (37.00 GHz H)

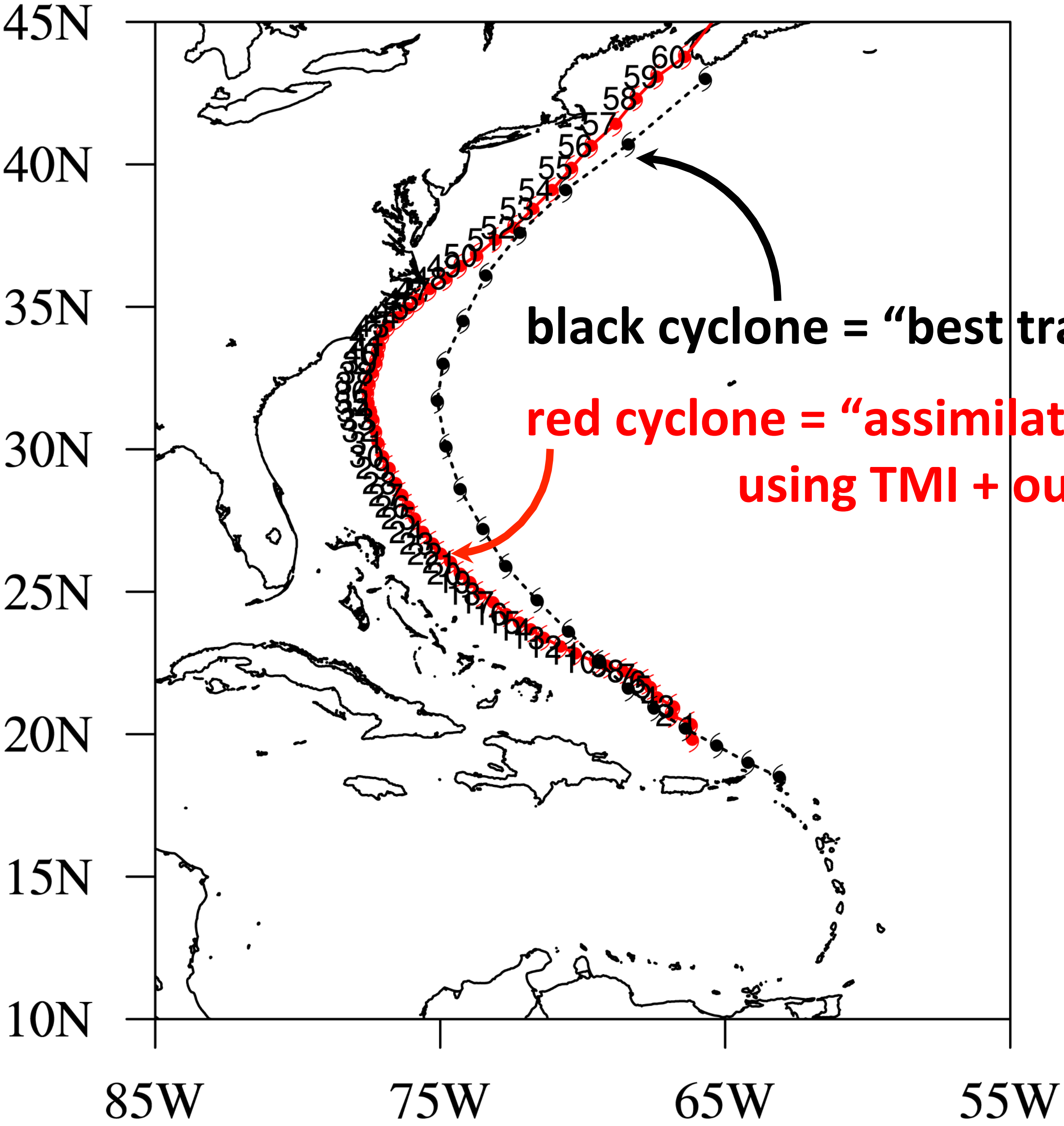

TB for TRMM ch 8 (85.00 GHz V)


TB for TRMM ch 9 (85.00 GHz H)

Step 4: EnKF assimilate - GRIP experiment (try #1, single 9km grid)



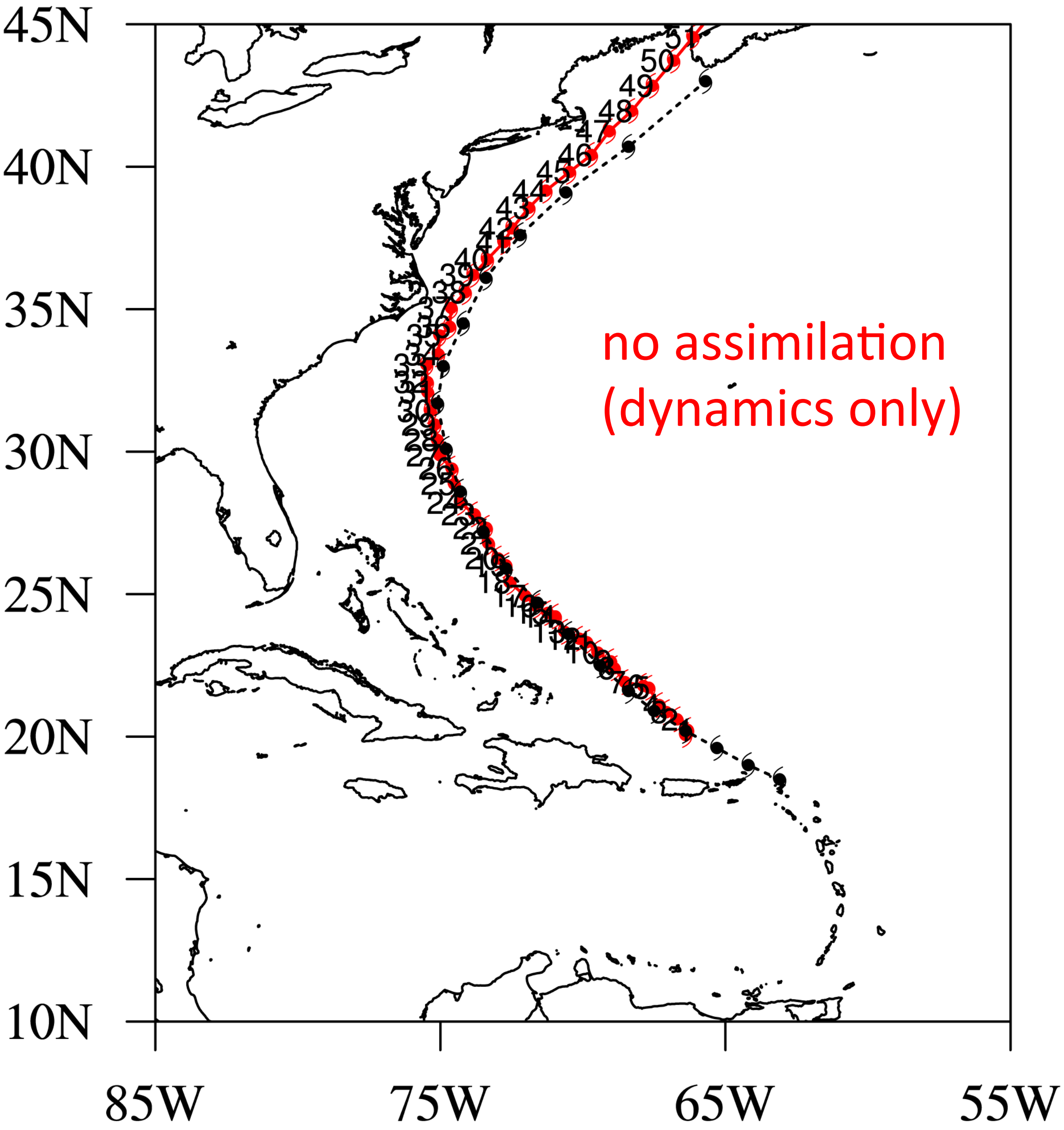
Step 4: EnKF assimilate - GRIP experiment (try #1, single 9km grid)



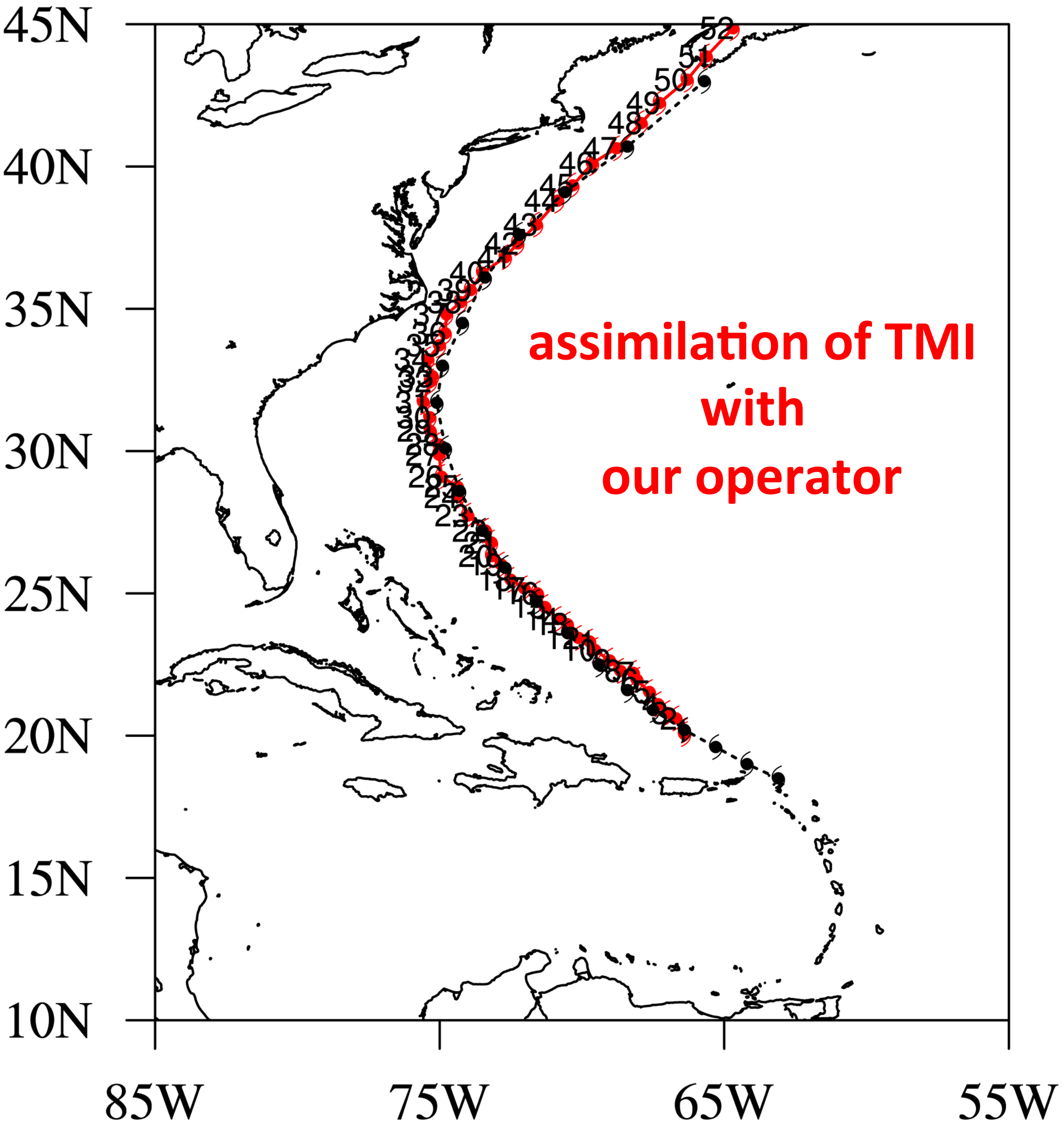
black cyclone = “best track”

red cyclone = “assimilated storm”, using TMI + our operator

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

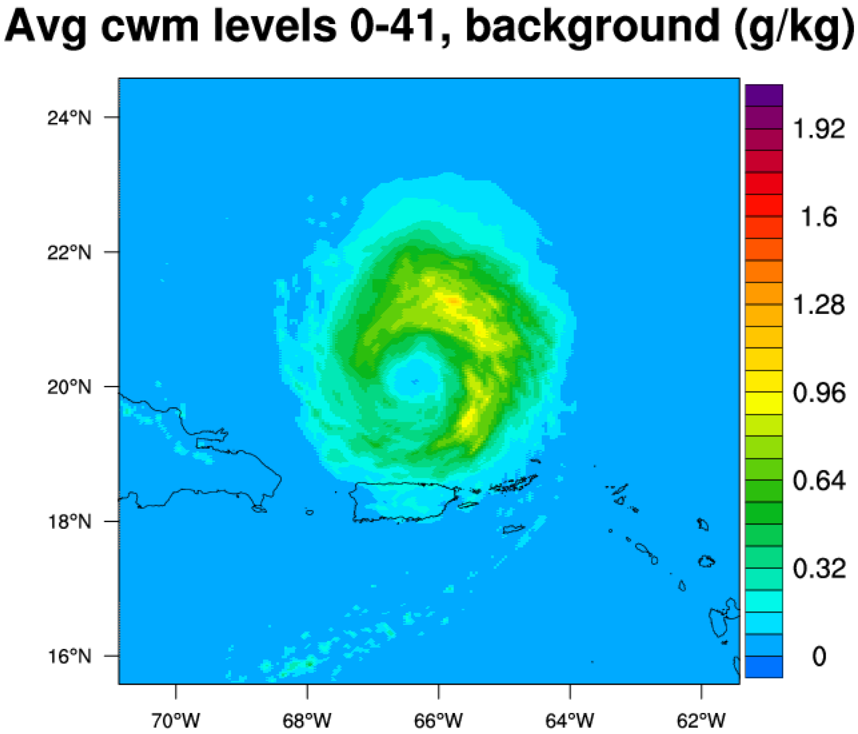


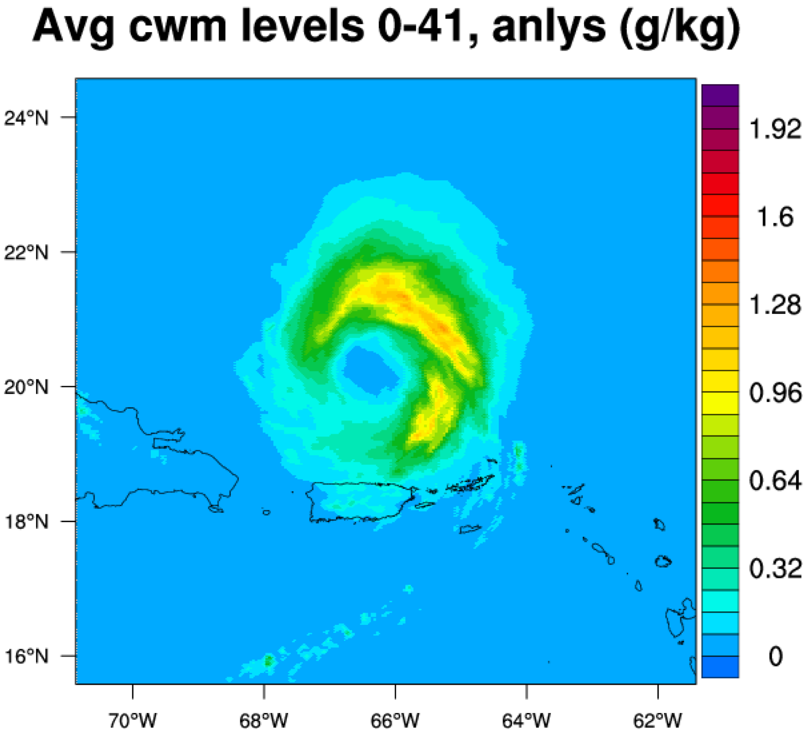
Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)



assimilation of TMI
with
our operator

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

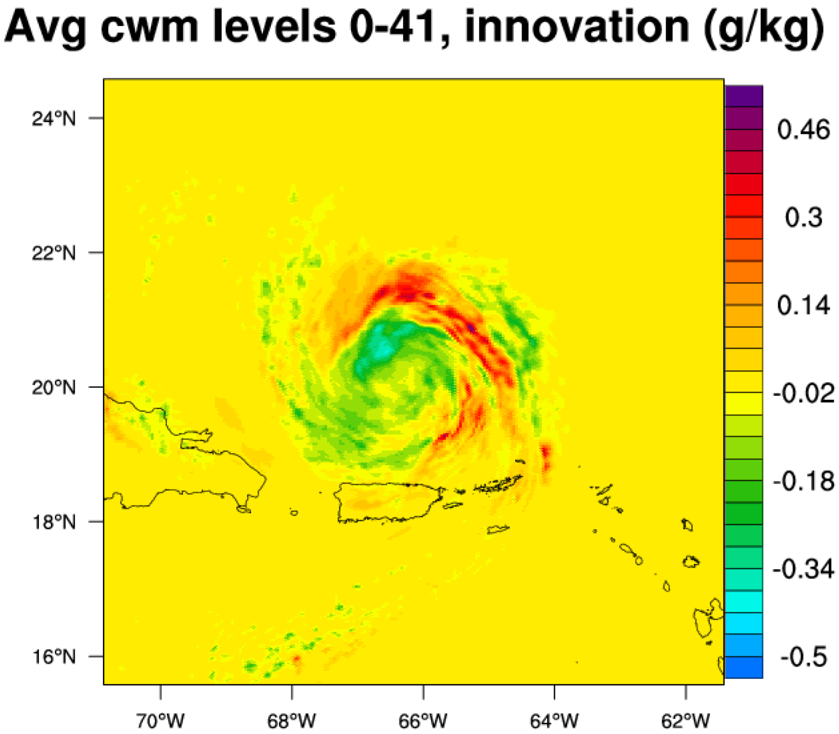




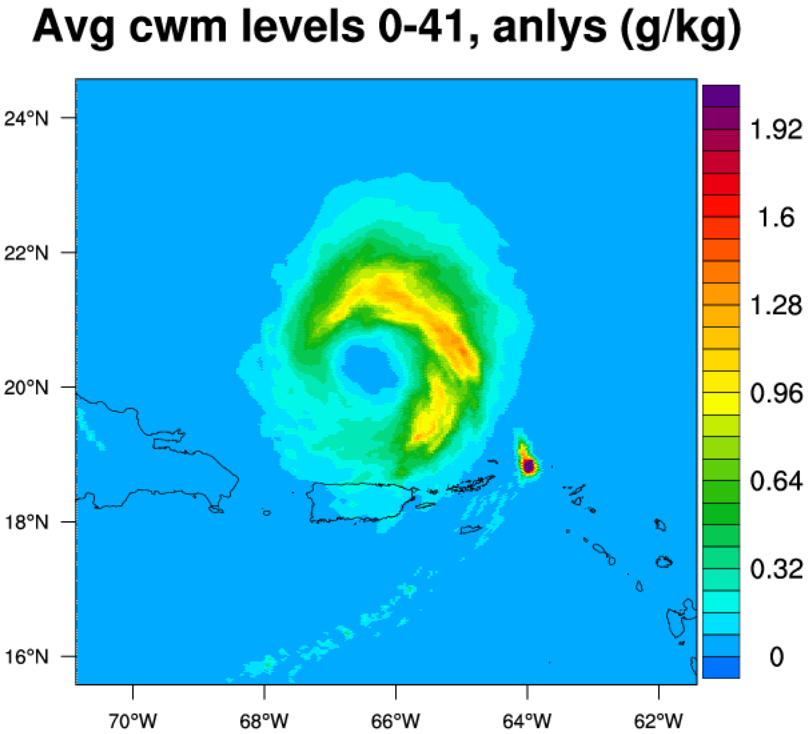
total condensed water "before"

total condensed water "after"

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

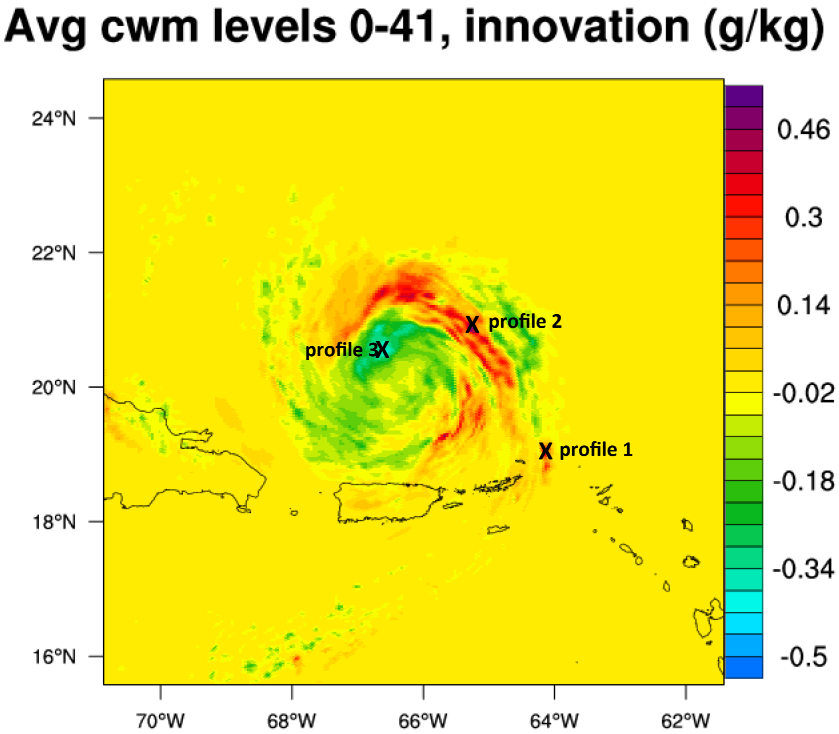


total condensed water innovation

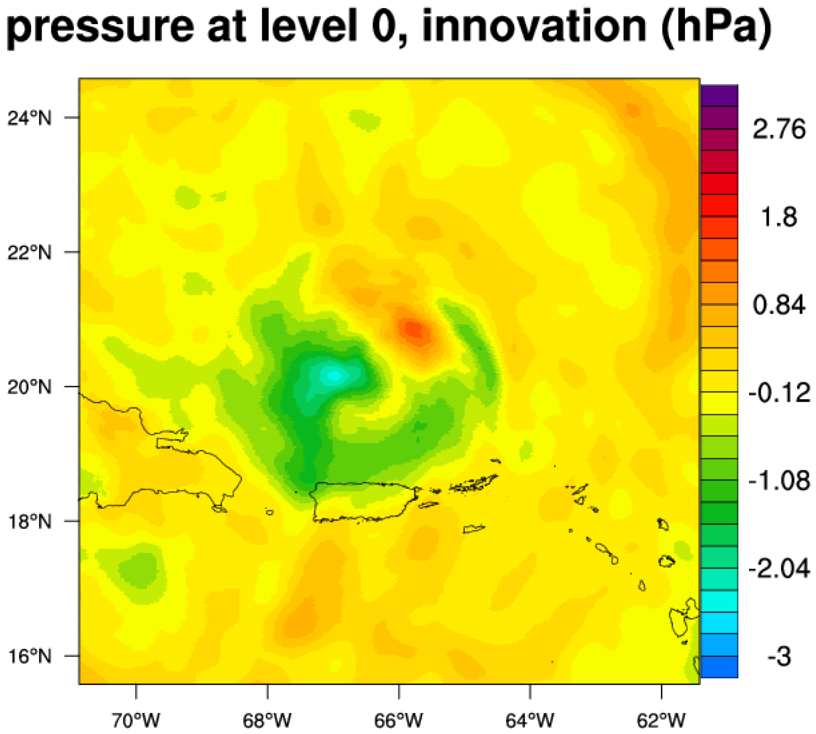


total condensed water “after” 4x smaller obs covariance

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

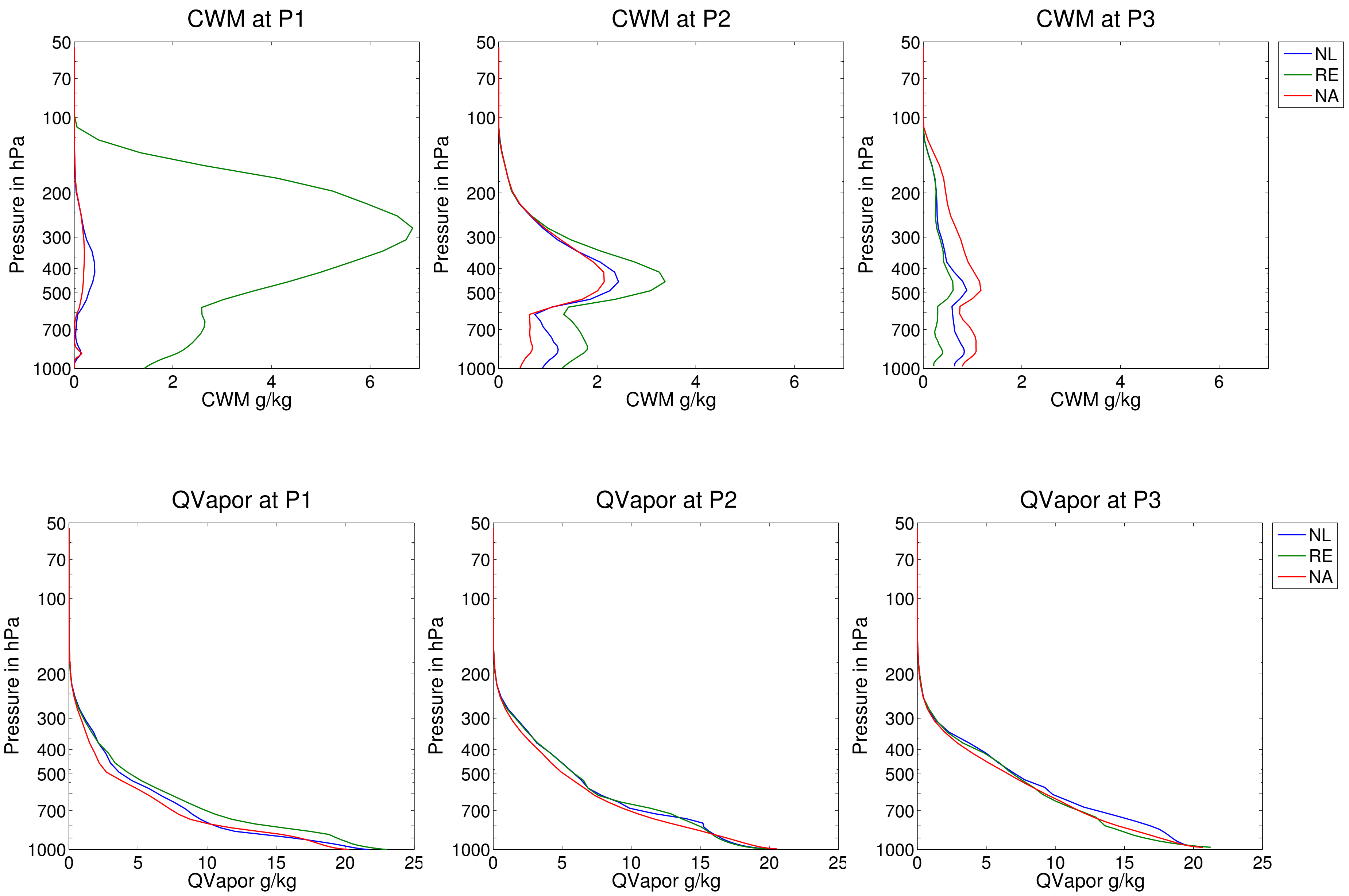


total condensed water innovation



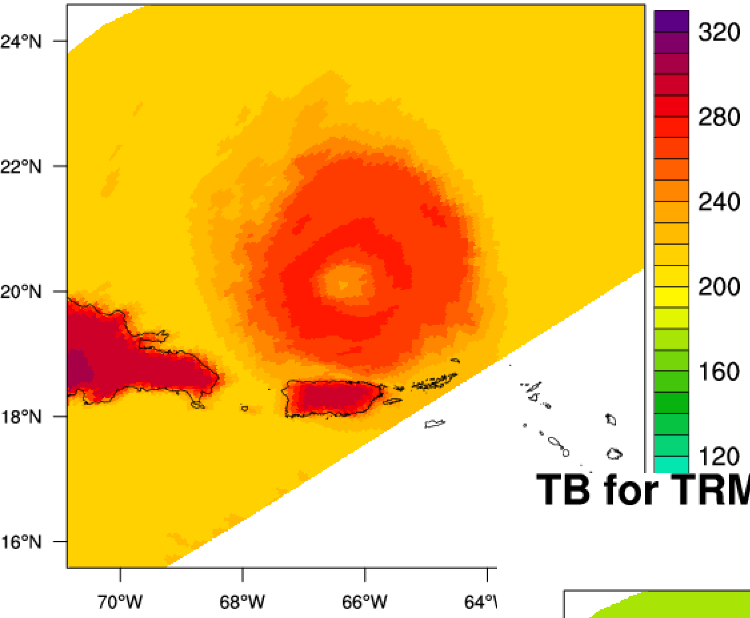
surface pressure innovation

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

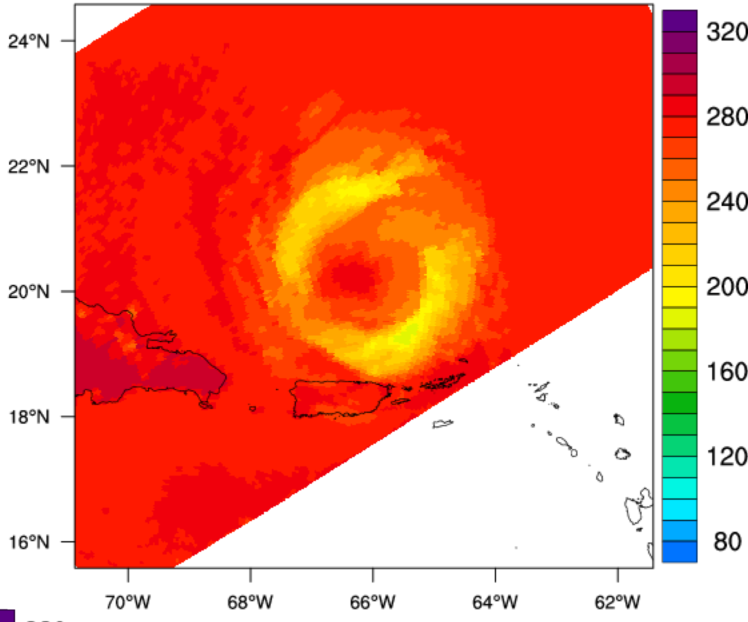


Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

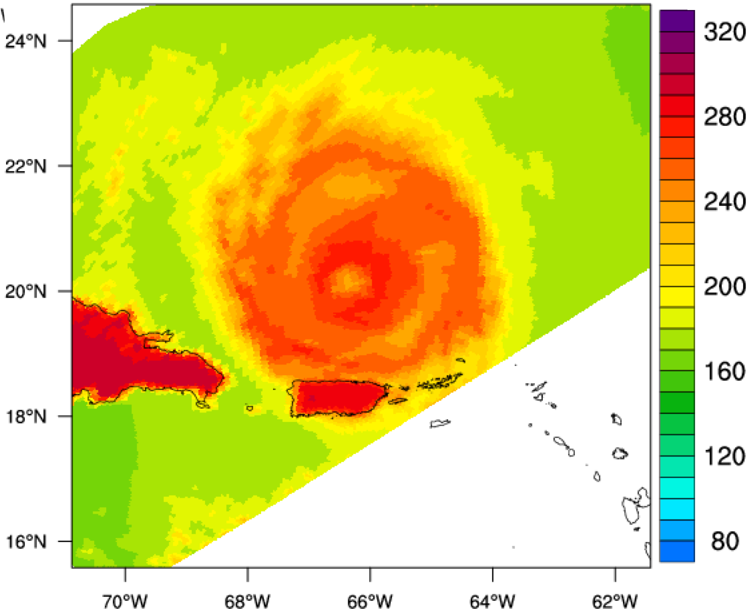
TB for TRMM ch 3 (19.35 GHz V)



TB for TRMM ch 8 (85.50 GHz V)

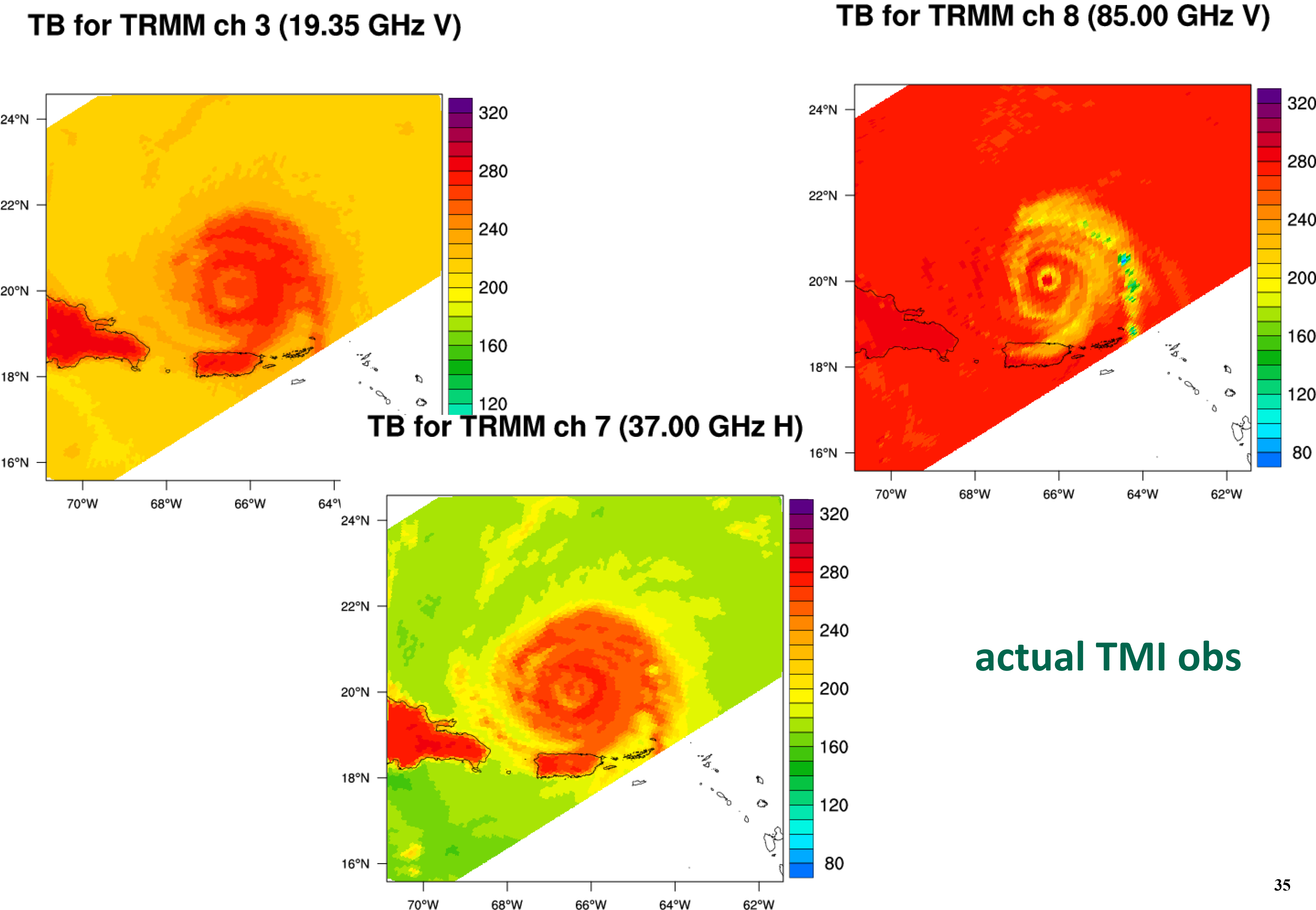


TB for TRMM ch 7 (37.00 GHz H)



forward-calc obs
from **no assimilation**

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)



TB for TRMM ch 3 (19.35 GHz V)

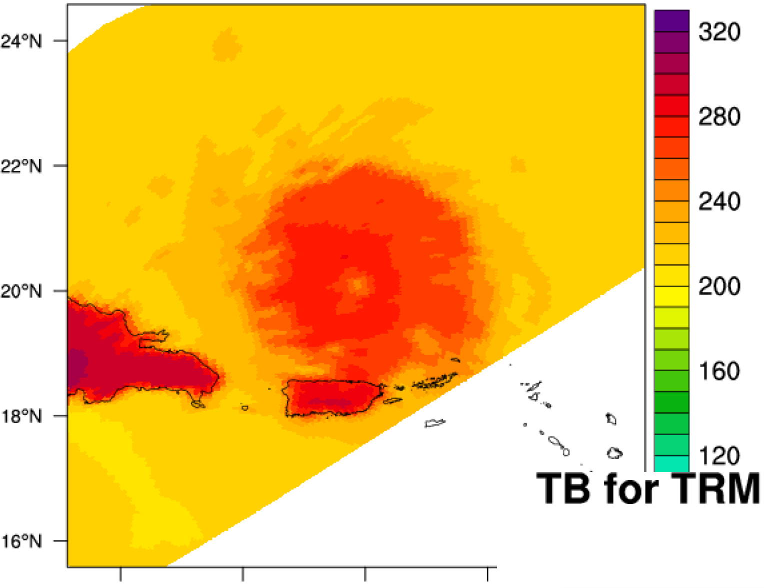
TB for TRMM ch 8 (85.00 GHz V)

TB for TRMM ch 7 (37.00 GHz H)

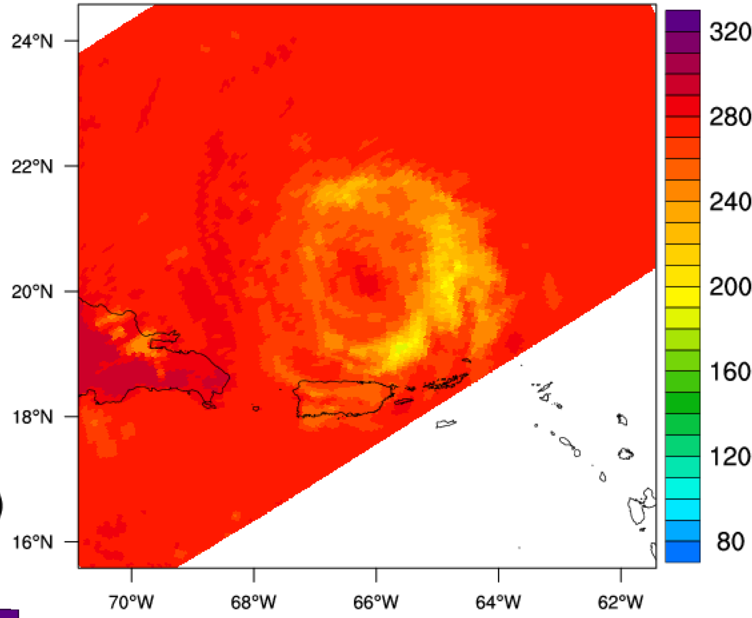
actual TMI obs

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

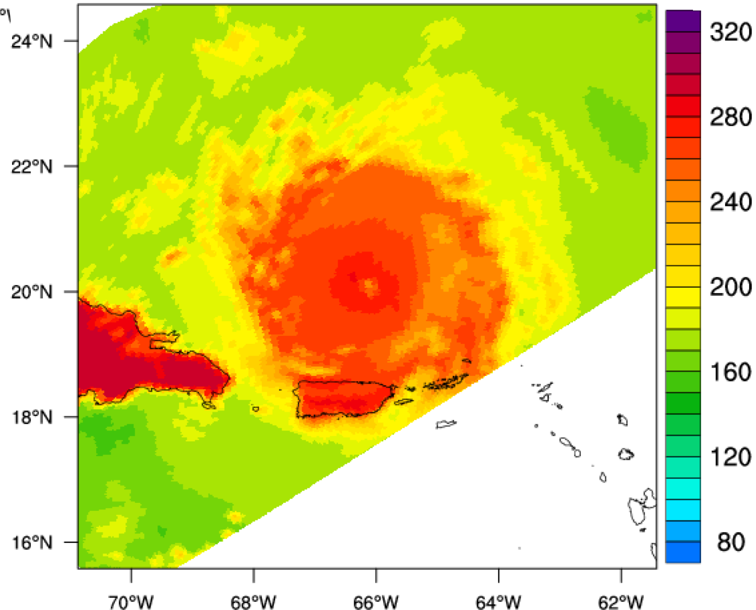
TB for TRMM ch 3 (19.35 GHz V)



TB for TRMM ch 8 (85.50 GHz V)



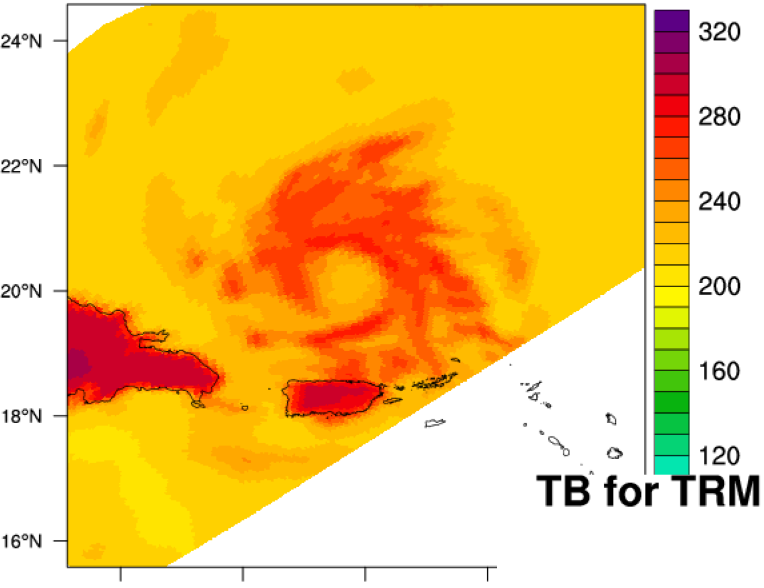
TB for TRMM ch 7 (37.00 GHz H)



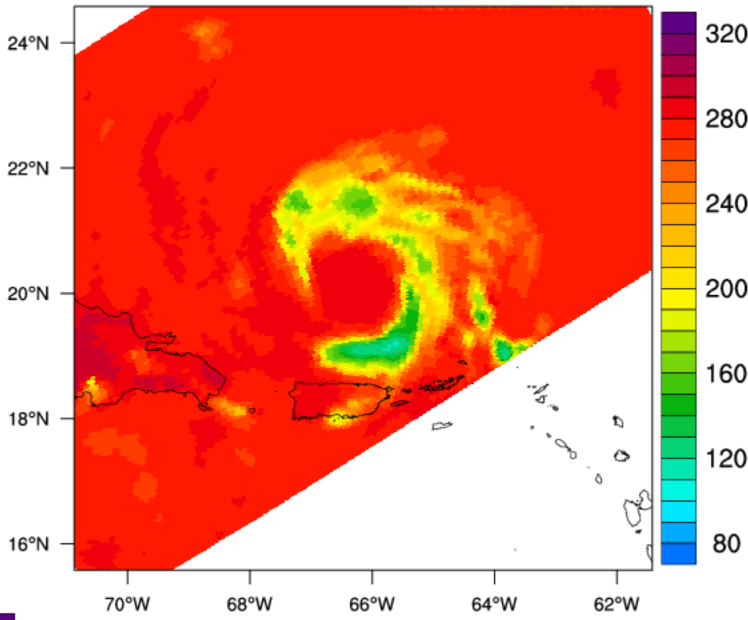
forward-calc obs
from **analysis**

Step 4: EnKF assimilate - GRIP experiment (try #2, nested grids)

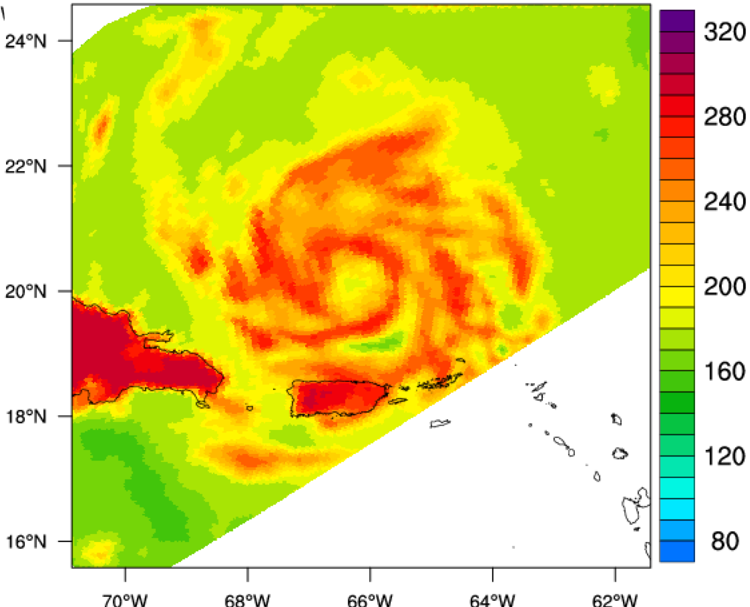
TB for TRMM ch 3 (19.35 GHz V)



TB for TRMM ch 8 (85.50 GHz V)



TB for TRMM ch 7 (37.00 GHz H)



forward-calc obs
from **analysis,**
1 hour later

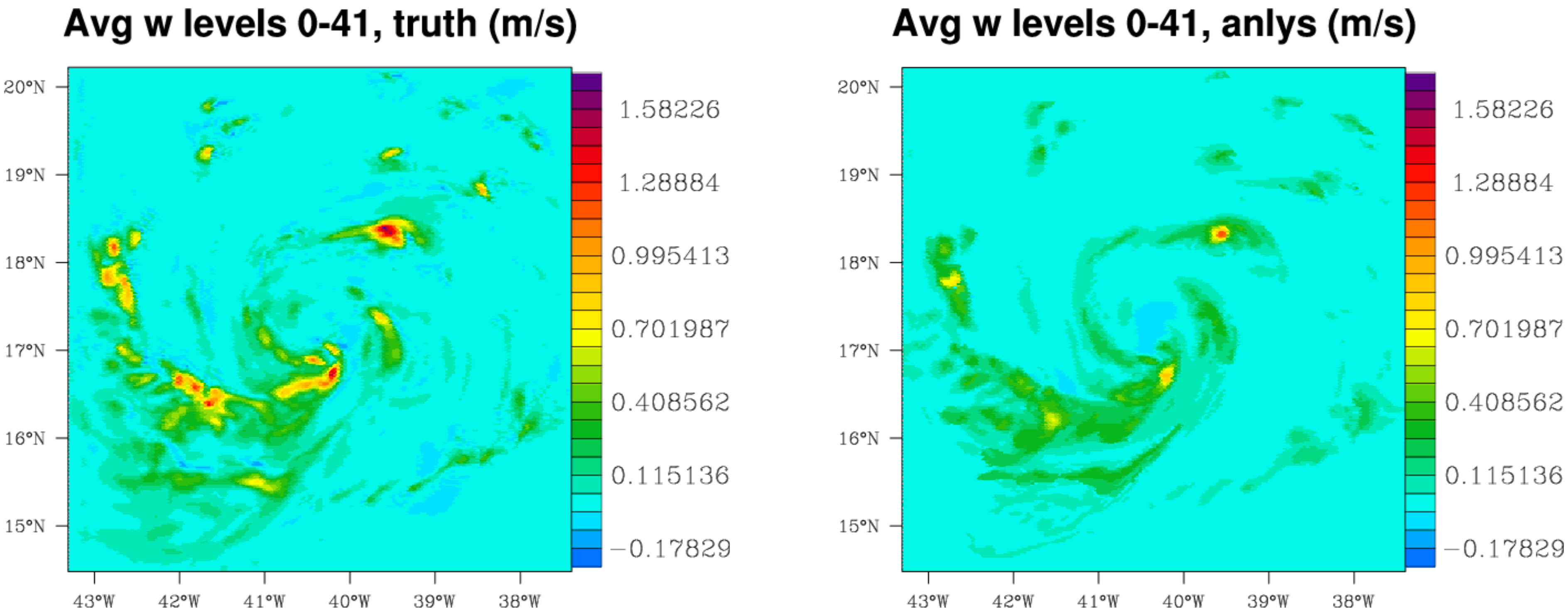
Same nonlinear operator, but applied to Igor (still with TMI):

$$T_i''(x_1'', x_2'', x_3'') = \Sigma \quad T_i''^{(n)} \exp(-[x_1''-x_1''^{(n)}]^2 -[x_2''-x_2''^{(n)}]^2 -[x_3''-x_3''^{(n)}]^2)$$

assimilation using this observation operator:

(start with 1D-var – not quite a “retrieval”, because of the covariances)

Having started with a horizontally uniform background,
each variable having the global mean value at that height level:



Avg w levels 0-41, truth (m/s)

Avg w levels 0-41, anlys (m/s)

vertical component of wind

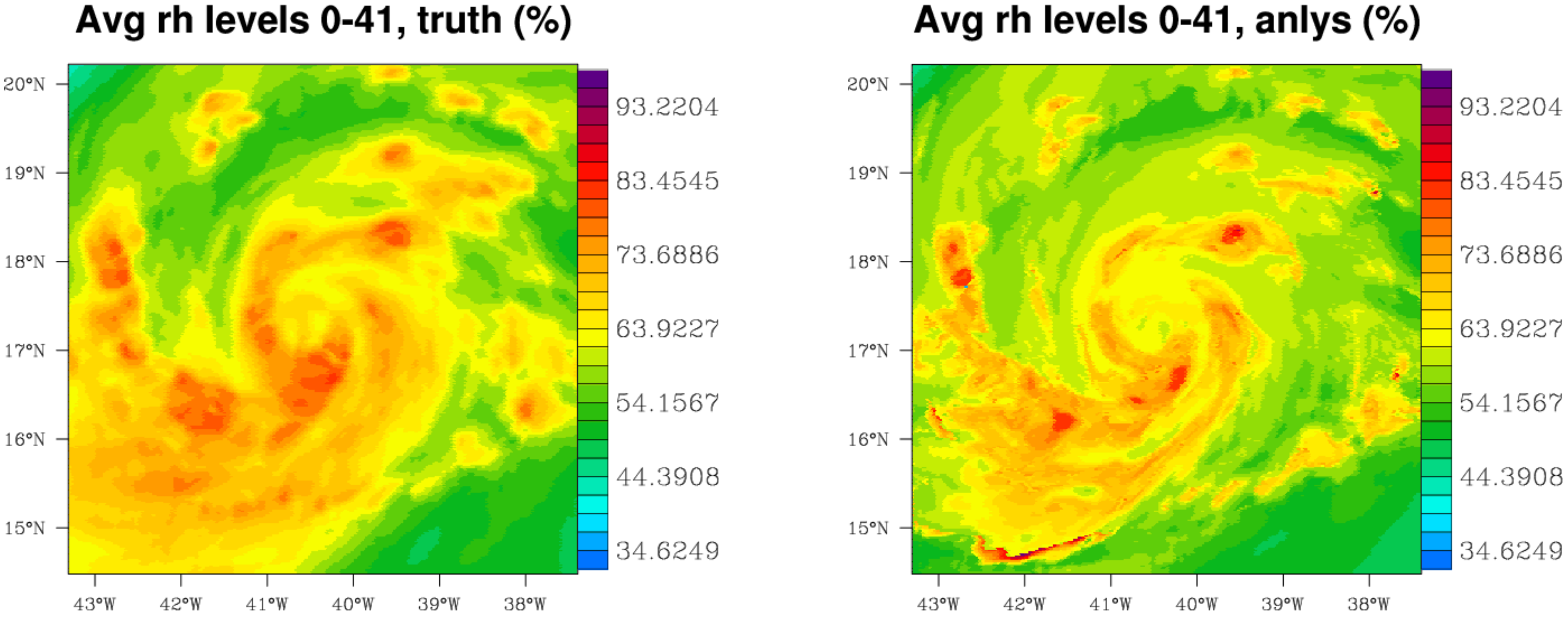
Same nonlinear operator, but applied to Igor (still with TMI):

$$T_i''(x_1'', x_2'', x_3'') = \Sigma \; T_i''^{(n)} \exp(-[x_1''-x_1''^{(n)}]^2 -[x_2''-x_2''^{(n)}]^2 -[x_3''-x_3''^{(n)}]^2)$$

assimilation using this observation operator:

(start with 1D-var – not quite a “retrieval”, because of the covariances)

Having started with a horizontally uniform background,
each variable having the global mean value at that height level:



Avg rh levels 0-41, truth (%)

Avg rh levels 0-41, anlys (%)

water vapor

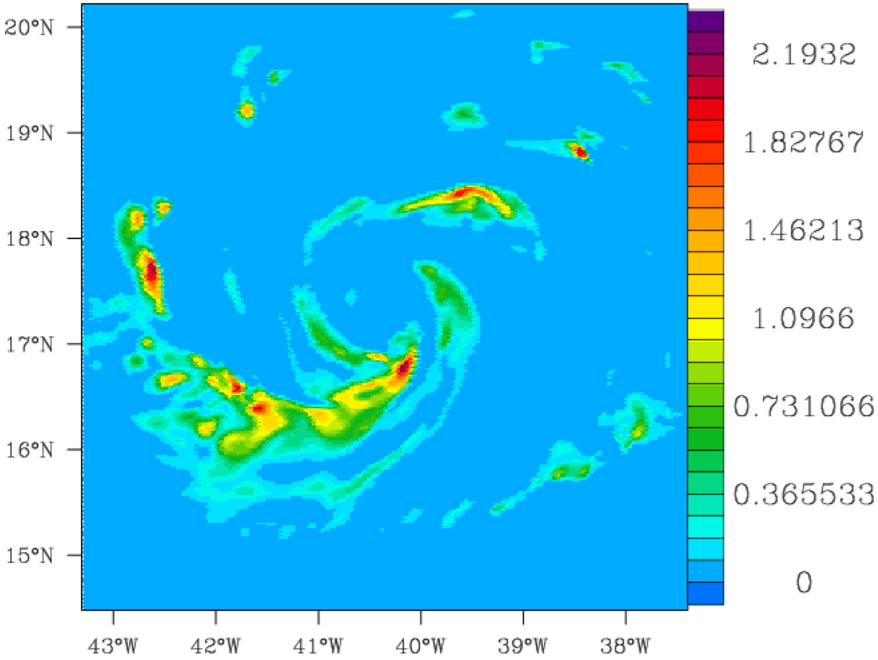
Same nonlinear operator, but applied to Igor (still with TMI):

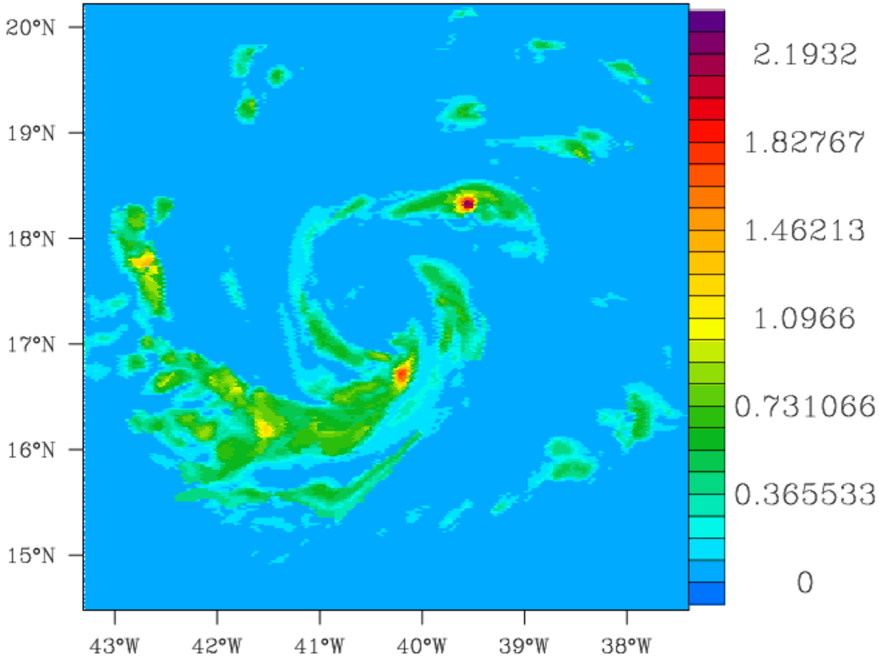
$$T_i''(x_1'', x_2'', x_3'') = \Sigma \; T_i''^{(n)} \exp(-[x_1''-x_1''^{(n)}]^2 -[x_2''-x_2''^{(n)}]^2 -[x_3''-x_3''^{(n)}]^2)$$

assimilation using this observation operator:

(start with 1D-var – not quite a “retrieval”, because of the covariances)

Having started with a horizontally uniform background,
each variable having the global mean value at that height level:


Avg qrain levels 0-41, truth (g/kg)


Avg qrain levels 0-41, anlys (g/kg)

rain

Same nonlinear operator, but applied to Igor (still with TMI):

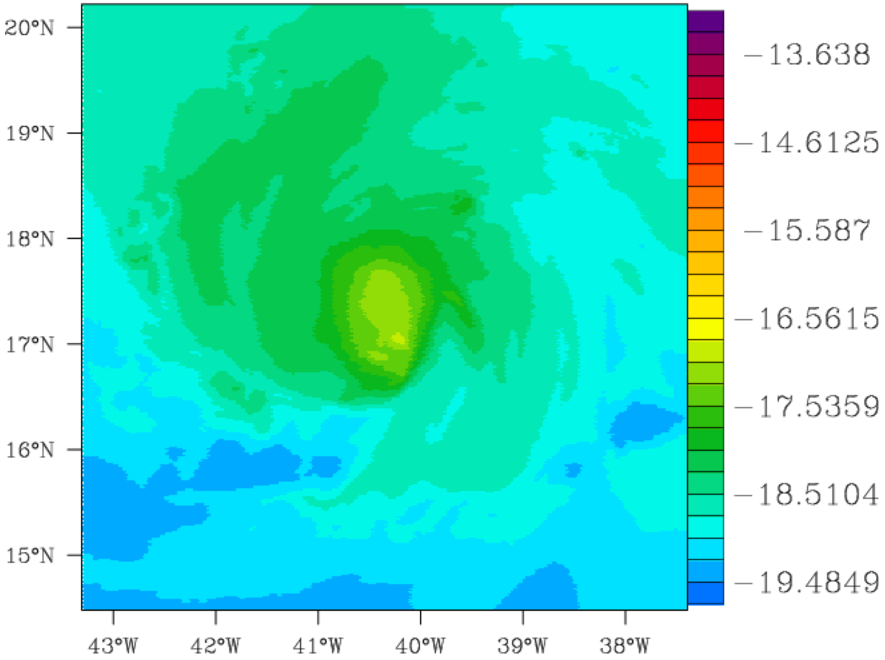
$$T_i''(x_1'', x_2'', x_3'') = \Sigma \quad T_i''^{(n)} \exp(-[x_1''-x_1''^{(n)}]^2 -[x_2''-x_2''^{(n)}]^2 -[x_3''-x_3''^{(n)}]^2)$$

assimilation using this observation operator:

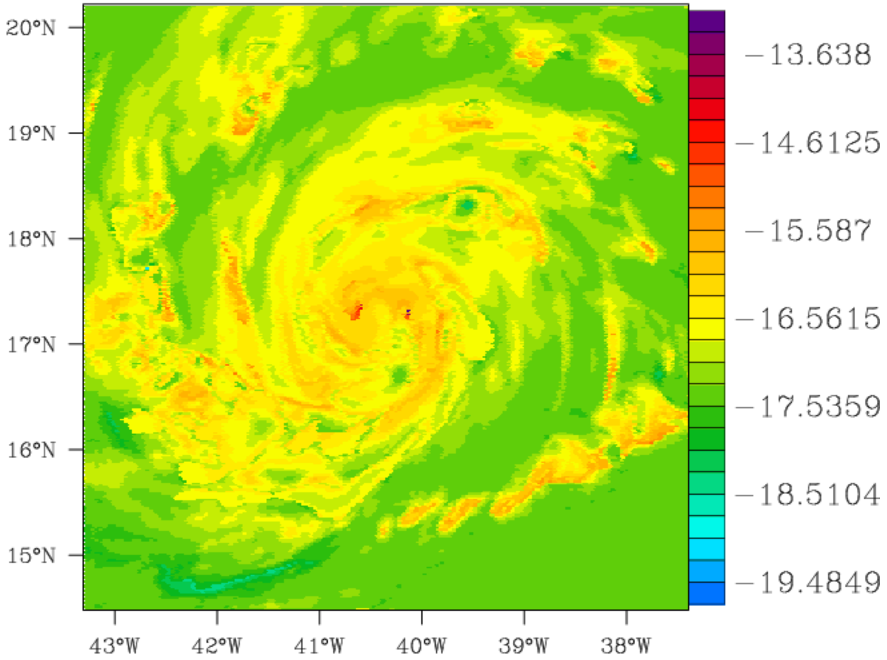
(start with 1D-var – not quite a “retrieval”, because of the covariances)

Having started with a horizontally uniform background,
each variable having the global mean value at that height level:

Avg temp levels 0-41, truth (C)



Avg temp levels 0-41, anlys (C)



temperature

still finds a warm core!
(but warm bias throughout,
balancing rain “under-emission”)