SWG3.2: Adapting RTTOV for ground-based microwave radiometers

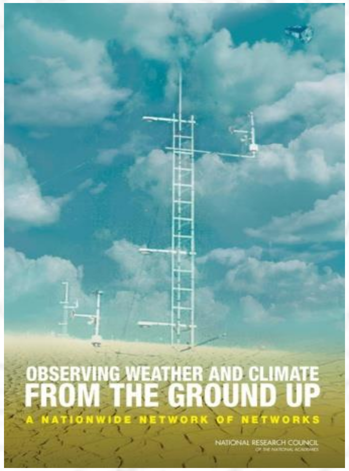
30 March – 1 April, 2015

D. Cimini[1,2], F. De Angelis[2], J. Hocking[3], P. Martinet[4]

[1] CNR-IMAA, Potenza, Italy, domenico.cimini@imaa.cnr.it
[2] CETEMPS, University of L'Aquila, Italy
[3] MetOffice, Exeter, UK
[4] Météo France, Toulouse, France

Motivations

- The PBL is the single most important under-sampled part of the atmosphere*
  - Surface → met data
  - Upper air → satellite → GAP in the PBL
- Particularly important in nowcasting and severe weather initiation
- Four critical atmospheric variables are not adequately measured**
  - wind profiles,
  - **temperature and humidity profiles (in cloudy areas),**
  - precipitation, snow mass.
- Ground-based MWR provide T and H profiles with:
  - High temporal resolution (~1 min),
  - Low-to-moderate vertical resolution,
  - Information mostly residing in the PBL.



*U.S.NRC Reports; **WMO guidance on observations for NWP

Motivations

- Recent experiments have demonstrated the feasibility and impact of MWR data assimilation (DA)
  - DA of temperature and humidity profiles
- Improvements are expected from DA of radiances (Tb)
  - A fast forward model is needed
- A fast forward model is available from NWP SAF (MetOffice) for satellite radiometric observations: RTTOV
  - Suitable for microwave radiometers
  - RTTOV can be adapted to work for ground-based obs

*https://www.wmo.int/pages/prog/www/OSY/GOS-RRR.html

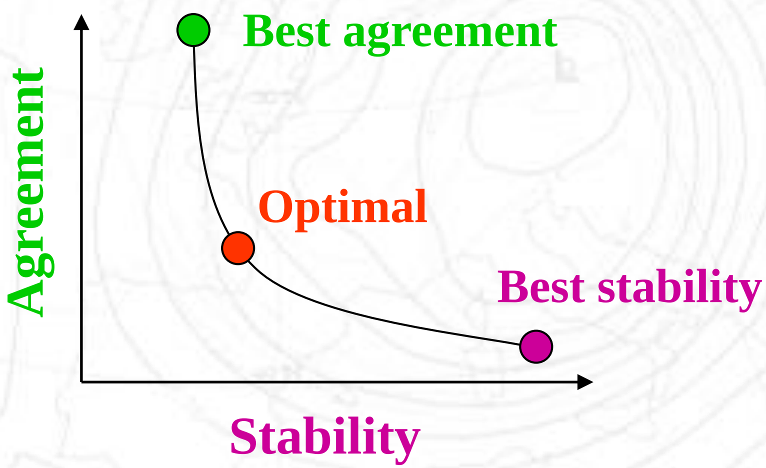
Variational approach (such as 3DVAR)

- Minimization of a cost function

$$J = \left[y - F(x)\right]^T R^{-1}\left[y - F(x)\right] + \left[x - x_b\right]^T B^{-1}\left[x - x_b\right]$$

$$x_{i+1} = x_i + \left(B^{-1} + K_i^T R^{-1} K_i\right)^{-1}\left[K_i^T R^{-1}\left(y - F(x_i)\right) - B^{-1}(x_i - x_b)\right]$$

- The solution requires
  - the forward model (radiative transfer operator) F(x)
  - the derivative of F(x), i.e Jacobian K, and its transpose K^T



Objectives of SWG3.2

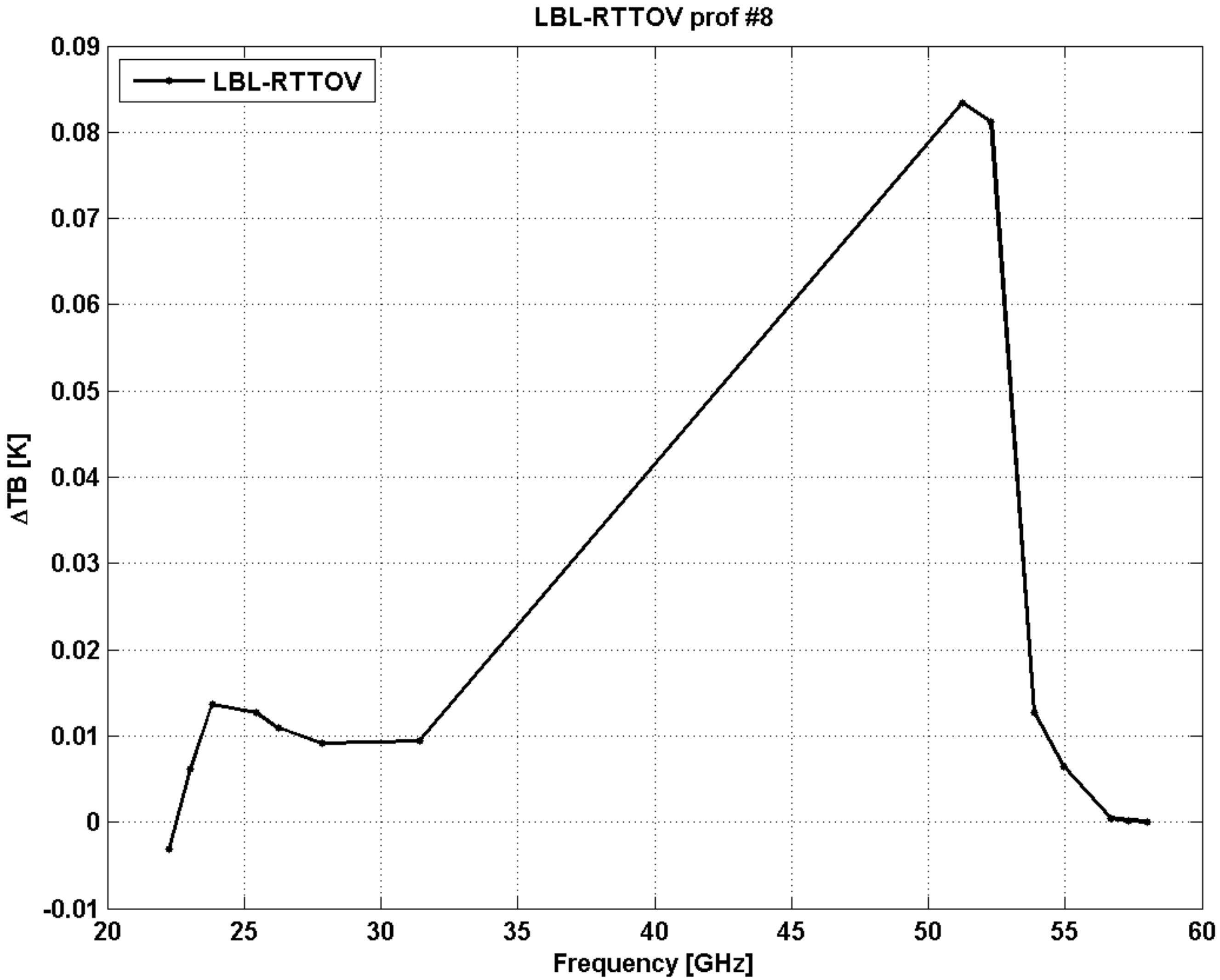
- Test the ground-based RTTOV (RTTOVgb) direct module
- Modify RTTOVgb to develop the K module
- Test the RTTOVgb K module
- Modify the 1-DVAR tool to handle multi-angle observations within the same observation vector;
- Provide RTTOVgb (including direct and K modules) for the implementation into the modified 1-DVAR scheme.

Testing RTTOVgb direct

- RTTOVgb direct was developed during a TOPROF STSM
    - Francesco De Angelis → MetOffice

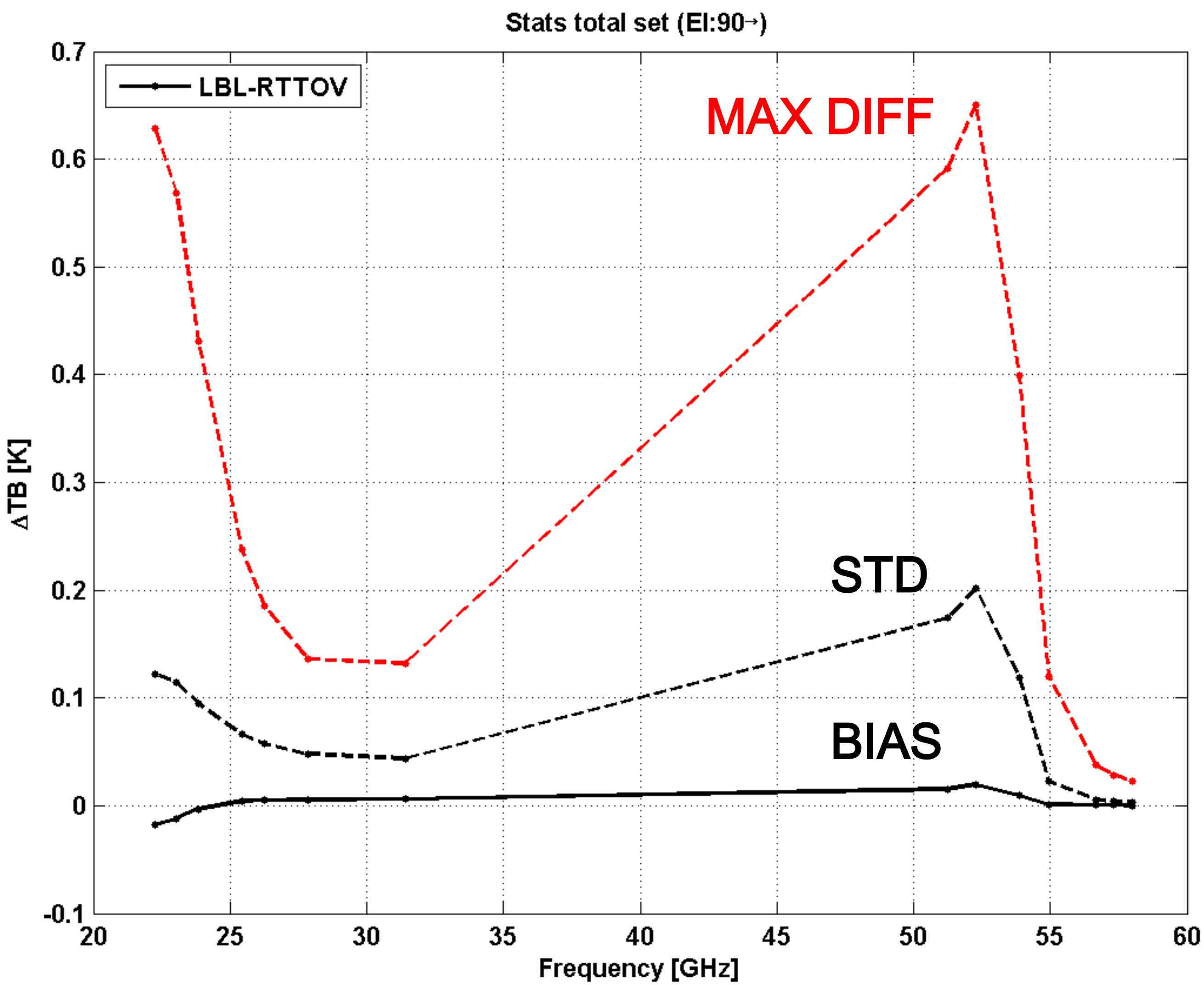
- RTTOVgb direct is tested against reference radiative transfer model output

Testing RTTOVgb direct



National Research Council of Italy

Testing RTTOVgb direct



Developing RTTOVgb K (Jacobian)

- The Jacobian (K) may be computed using finite differences
  - This requires 2N+1 forward model calculations
    - (N: number of levels)
    - "Brute force" method

Developing RTTOVgb K (Jacobian)

- Only the direct and K are needed

- But this implies also the tangent linear (TL) and the adjoint (AD) modules to assure a testing path

$$
K(x)^T = \begin{bmatrix}
\frac{\partial R_1}{\partial T_1} & \frac{\partial R_2}{\partial T_1} & \frac{\partial R_3}{\partial T_1} & \cdots & \frac{\partial R_m}{\partial T_1} \\
\frac{\partial R_1}{\partial T_2} & \frac{\partial R_2}{\partial T_2} & \frac{\partial R_3}{\partial T_2} & \cdots & \frac{\partial R_m}{\partial T_2} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{\partial R_1}{\partial T_n} & \frac{\partial R_2}{\partial T_n} & \frac{\partial R_3}{\partial T_n} & \cdots & \frac{\partial R_m}{\partial T_n} \\
\frac{\partial R_1}{\partial q_1} & \frac{\partial R_2}{\partial q_1} & \frac{\partial R_3}{\partial q_1} & \cdots & \frac{\partial R_m}{\partial q_1} \\
\frac{\partial R_1}{\partial q_2} & \frac{\partial R_2}{\partial q_2} & \frac{\partial R_3}{\partial q_2} & \cdots & \frac{\partial R_m}{\partial q_2} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{\partial R_1}{\partial q_n} & \frac{\partial R_2}{\partial q_n} & \frac{\partial R_3}{\partial q_n} & \cdots & \frac{\partial R_m}{\partial q_n}
\end{bmatrix}
$$

- Developing chain: TL is derived from the direct → AD is derived from the TL → K is derived from AD

direct → TL → AD → K

From direct to TL

- TL code is derived from the direct code
- TL is the analytic derivative of the radiance (F(x)) wrt the state vector x M-D vector, M: number of channels

Tangent Linear Coding Example

Equation:

$$y = a + bx + cx^2 + dx^3 + ez^{1/2}$$

Differential:

$$y' = bx' + 2cxx' + 3dx^2x' + 1/2ez^{-1/2}z'$$

Forward Code:

```
Y = A + B*X + C*X**2. + D*X**3. + E*SQRT(Z)
```

TL Code:

```
Y_TL = B*X_TL + 2.*C*X*X_TL + 3.*D*X**2.*X_TL  &
     * .5*E*Z**(-.5)*Z_TL
```

Testing TL module

Testing TL Model consistency with Forward Model

$$\lim_{\Delta \mathbf{x}^{\pm} \to 0} \frac{FM(\mathbf{x} + \Delta \mathbf{x}) - FM(\mathbf{x})}{TLM(x, \Delta \mathbf{x})} = 1$$

This looks a lot like the definition of the derivative.

FM(x) = Forward model acting on x
FM(x+Δx)=perturbed Forward model acting on x+Δx
TL(x,Δx)= Tangent Linear model acting on Δx (at x)

From TL to AD

- AD is the transpose of the derivative of the radiance (F(x)) wrt the state vector x N-D vector, N: number of levels

- AD is derived from the TL by taking the transpose of the TL code

```
TL
Real Function Bright_TL
(V,Radiance,Radiance_TL,BC1,BC2)
K2 = C2*V
K1 = C1*V*V*V

TempTb_TL =
    K2*Alog(K1/Radiance + 1.)**(-2.)
    * Radiance_TL/(K1+Radiance) *
    K1/Radiance        (1)

Bright_TL = BC2*TempTb_TL      (2)

 Return
End Function Bright_TL
```

```
Adjoint
Subroutine Bright_AD
    (V,Radiance,Radiance_AD,BC1,
    BC2,TB_AD)
K2 = C2*V
K1 = C1*V*V*V  !inactive constants
TempTb_AD = 0 ! initialize for each
    invocation
TempTb_AD = TempTb_AD +
    BC2*Tb_AD      (2)

Radiance_AD = Radiance_AD +
   K2*Alog(K1/Radiance + 1.)**(-2.)
    * TempTb_AD/(K1+Radiance) *
    K1/Radiance      (1)
 Return
End Subroutine Bright_AD
```

From AD to K

- K code development is the easiest of the three
- Distribute the AD level derivatives through the number of channels

Testing TL, AD, K consistency

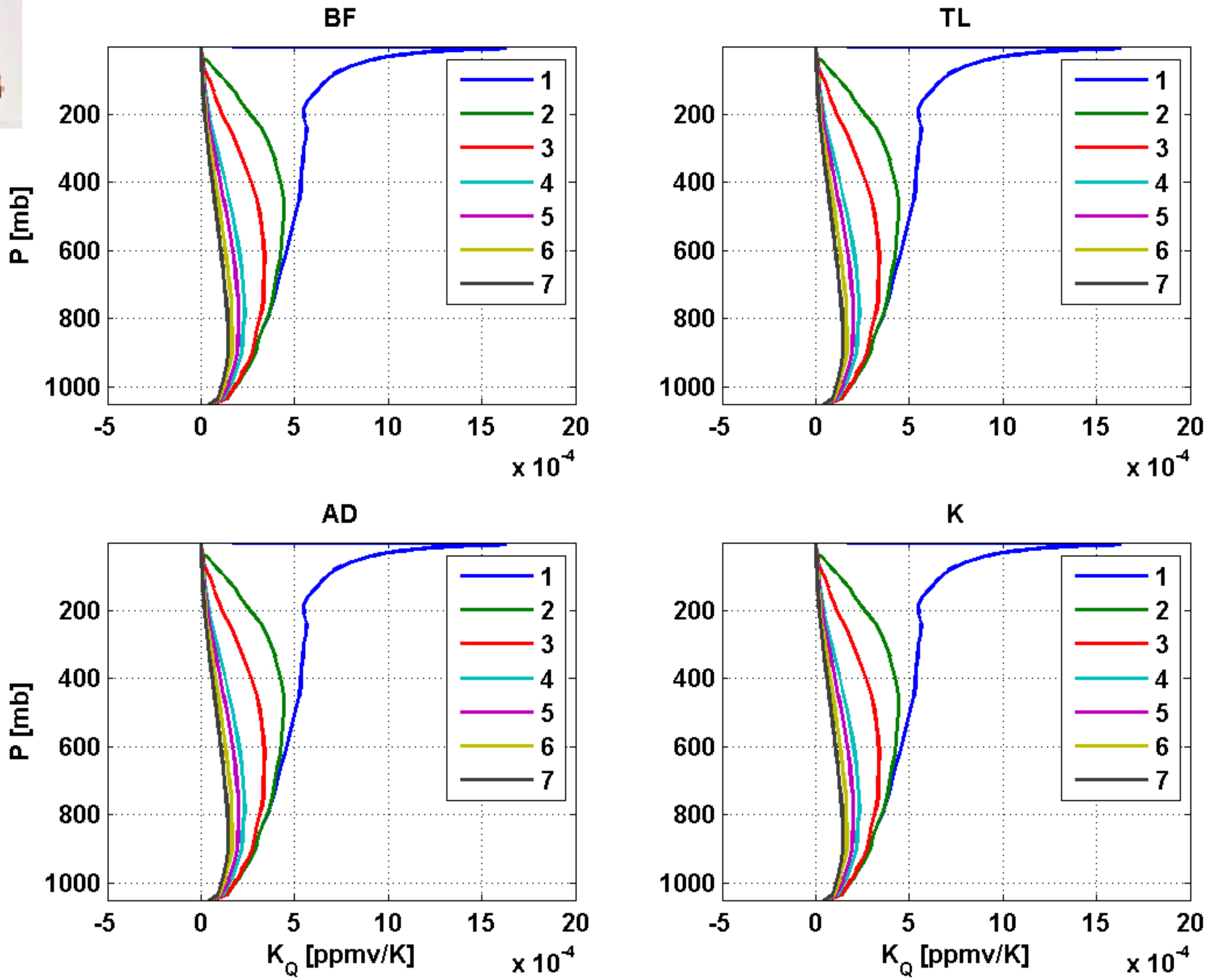
- **Method**: Construct K from TL and AD and compare
- K_TL, K_AD and K should be exactly the same to within machine precision
- We used the testing suite developed at NWP SAF for RTTOV

Four ways to compute Jacobians

	calculations
1. Brute Force (finite differences)	2N+1
2.Tangent Linear	N
3. Adjoint	M
4. K	~ 3

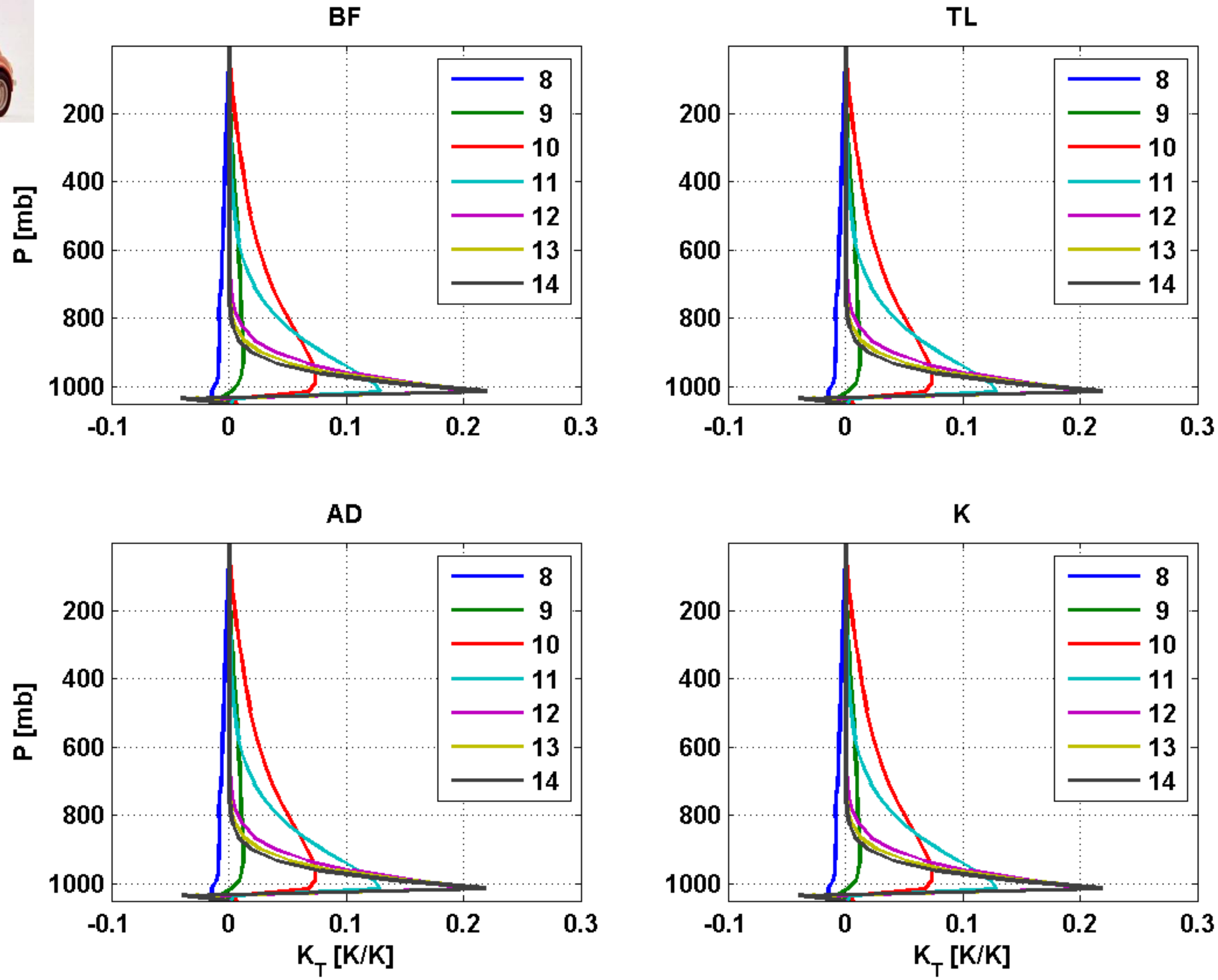
Testing RTTOVgb K consistency

HUMIDITY



Testing RTTOVgb K consistency

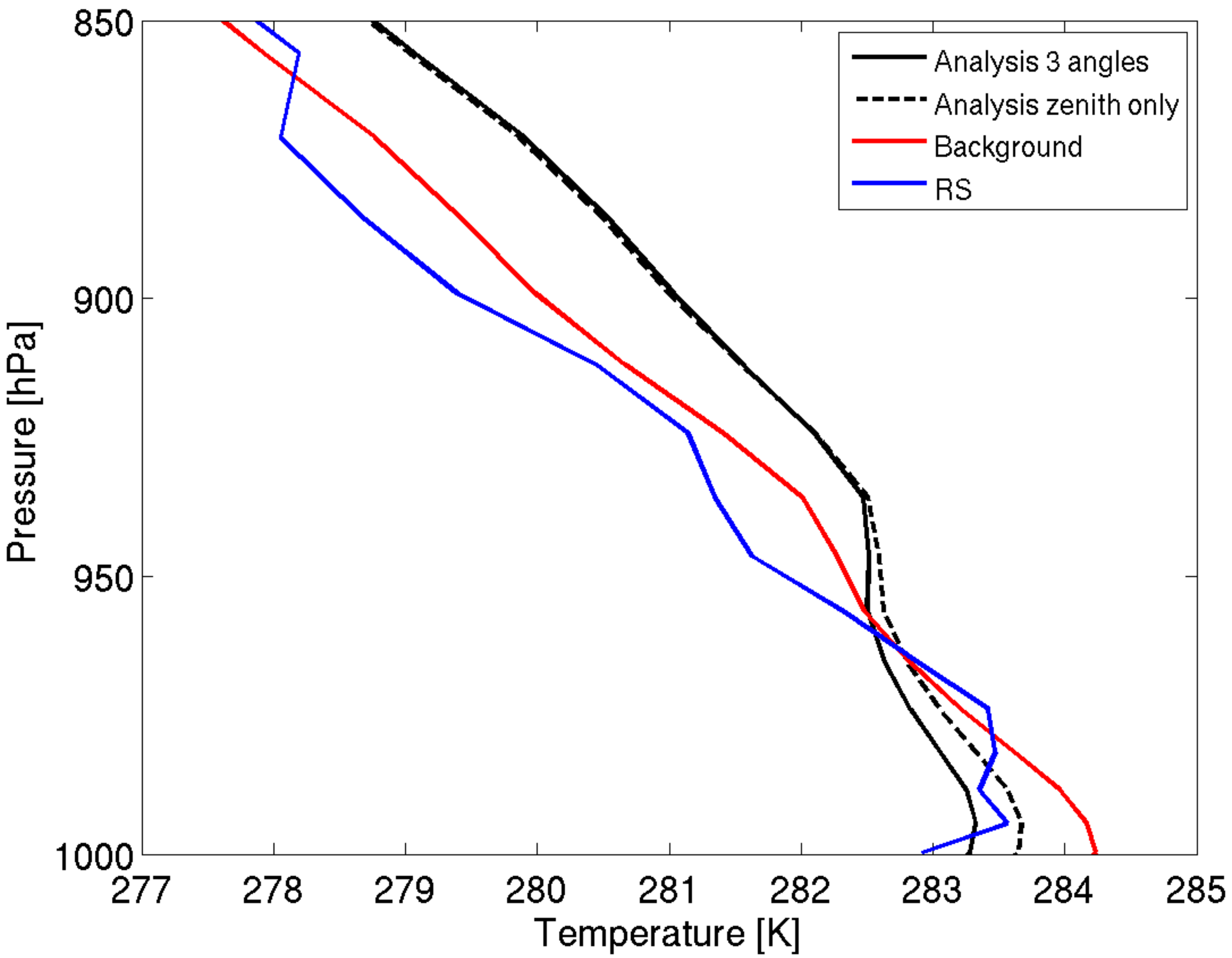
TEMPERATURE



Using RTTOVgb into 1DVAR

- Start from the 1-DVAR tool package developed at NWP SAF
- Modified to work with the RTTOVgb
- However, the package does not allow different elevation angles within the same observation vector
- The software package was modified to allow the exploitation of a composite observation vector y_o consisting of observations from two or more independent instruments.
- The "composite instrument" was modified to exploit a composite observation vector y_o consisting of observations from the same instrument but at different elevation angles.

Using RTTOVgb into 1DVAR



CONCLUSIONS

- Direct module was tested against reference calculations
  - Bias < 0.02 K   Std/rms < 0.2 K   Max < 0.7 K
- TL, AD, and K modules have been developed for RTTOVgb
- TL, AD and K modules were successfully tested
- The NWP SAF 1-DVAR tool was modified to handle multi-angle observations within the same observation vector
- The current version of RTTOVgb is currently being tested within the 1-DVAR tool

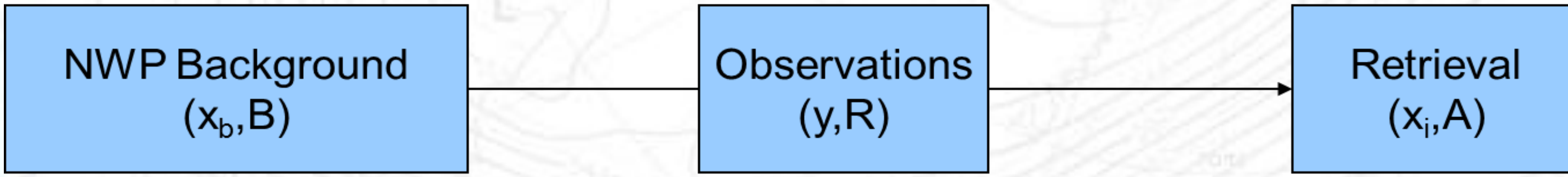
Remaining work

- Further testing of RTTOVgb to reduce differences with respect to reference radiative transfer models;
- Modify to allow lower elevation angles (now limited to 30°)
- Test 1-DVAR results with colocated radiosonde profiles (or other available reference) to demonstrate the performances

Thank you very much for your attention!!

One Dimensional Variational Assimilation Retrieval (1DVAR)

Nonlinear retrieval technique based on **Optimal Estimation Method** with a first guess taken from a **Numerical Weather Prediction** (NWP) model output



Assumptions: Moderately non-linear problem, Gaussian-distributed errors
Method: Gauss-Newton (Newtonian iteration with small residuals)

Cost function* **(to be minimized)**

$$J = [\mathrm{y} - F(\mathrm{x})]^{\mathrm{T}} R^{-1} [\mathrm{y} - F(\mathrm{x})] + [\mathrm{x} - \mathrm{x_b}]^{\mathrm{T}} B^{-1} [\mathrm{x} - \mathrm{x_b}]$$

Iterative solution

$$\mathrm{x_{i+1}} = \mathrm{x_i} + \left(B^{-1} + K_i^{\mathrm{T}} R^{-1} K_i\right)^{-1} \left[K_i^{\mathrm{T}} R^{-1} \left(\mathrm{y} - F(\mathrm{x_i})\right) - B^{-1}(\mathrm{x_i} - \mathrm{x_b})\right]$$

Error covariance

$$A = \left(B^{-1} + K_i^T R^{-1} K_i\right)^{-1}$$

Convergence criterium

$$[F(\mathrm{x_{i+1}}) - F(\mathrm{x_i})]^{\mathrm{T}} S^{-1} [F(\mathrm{x_{i+1}}) - F(\mathrm{x_i})] << n(obs)$$

$$S = R\left(R + K_i B K_i^T\right)^{-1} R$$

***Hewison, 2007 - Standard notation of Ide et al. 1997**

Radiative Transfer Equation

- SATELLITE

$$L_{clr} = \tau_s * \varepsilon_s * B(T_s) + \int_{\tau_s}^{1} B(T)d\tau + (1-\epsilon_s) * {\tau_s}^2 * \int_{\tau_s}^{1} \frac{B(T)}{\tau^2}$$

Surface to space transmittance

Level to space transmittance

- GROUND

$$L_{clr} = \tau_{toa} * B(T_{BKG}) + \int_{\tau_{toa}}^{1} B(T)d\tau$$

(A) (B)

Space to surface transmittance

Level to surface transmittance

(A) = Cosmic term;

(B) = Atmospheric term.

RTE: Ground-based case

(A) $L_{COSMIC} = \tau_{LEV=1} * B(T_{BKG})$ with $\tau_{LEV=1} = \tau_{toa} = \tau_{i=1}$

(B) $L_{ATM} = \int_{\tau_{LEV=1}}^{\tau_{LEVSURF-1}} B(T)d\tau + surf.term = \sum_{i=NLEV-1}^{1}(\int_{\tau_i}^{\tau_{i+1}} Bd\tau) + \mathrm{s.t.}$

$$\int_{\tau_i}^{\tau_{i+1}} Bd\tau = \tau_{i+1}B_{i+1} - \tau_i B_i + \frac{1}{OD_{singlelay}(i)}\Delta B_i \Delta\tau_i =$$

$$= \Delta\tau_i * [B_{i+1} + \Delta B_i \frac{1}{OD_{singlelay}(i)}] - \tau_i \Delta B_i$$

For Ground-Based:

$$\begin{cases} \Delta\tau_i = \tau_{i+1} - \tau_i \\ \Delta B_i = B_i - B_{i+1} \\ OD_{singlelay}(i) = OD_i - OD_{i+1} \end{cases}$$

Developing RTTOVgb K (Jacobian)

- Only the direct and K are needed
  - But this implies also the tangent linear (TL) and the adjoint (AD) modules are needed to assure a testing path
- TL is the analytic derivative of the radiance wrt the state vector (M-D vector, M: number of channels)
- AD is the transpose of the derivative of the radiance wrt the state vector (N-D vector, N: number of levels)
- K is the transpose of the derivative of the radiance wrt the state vector by channel (N*M matrix)
- Developing chain: TL is derived from the direct → AD is derived from the TL → K is derived from AD
- TL code inputs linearized temperature profile and outputs linearized brightness temperature
- AD code inputs linearized brightness temperatures and outputs linearized temperature profile