The Community Radiative Transfer Model Technical Overview

CRTM Team:
Benjamin T. Johnson (Project Lead, UCAR/JCSDA)
Patrick Stegmann (UCAR / JCSDA)
Cheng Dang (UCAR/JCSDA)

Jim Rosinski, Yingtao Ma, Ming Chen, Isaac Moradi, Haixia Liu, Nick Nalli, Cory Martin, Daniel Abdi, Tom Greenwald, Emily Liu, Barbara Scherllin-Pirscher, Quanhua ”Mark” Liu, Sarah Lu, Ping Yang, Will McCarty, Bryan Karpowicz, Yanqiu Zhu, *and many others*.

CRTM Project Highlights

CRTM v2.4.0 publicly released, tested, delivered (within 3 months). public domain license

CRTM v3.0.0-beta code (Q. Liu) initial integration into JCSDA github; merge with 2.4.1 code in progress.

Stand-alone **Transmittance coefficient generation package** for MW and IR sensors completed, documented.

Microphysics-consistent cloud tables received from PSU, coordination on testing and evaluation

Full **netCDF4 support for Clouds and Aerosols complete**, transmittance expected this year

Suite of **regression, unit, and application tests** for standalone v2.4 (works with v2.3)

New ecCodes and C-make build systems replacing legacy Autotools

2 core-team **publications** in preparation (Transmittance and v2.4 release)

CRTM v2.4.0

CRTM v2.4.0

https://github.com/JCSDA/crtm

- **Released October 28, 2020**
- **New Features:**
  1. Support for netCDF4 file format reading: CloudCoeff.nc4 and AerosolCoeff.nc4
  2. OpenMP optimization (vs. profiles)
  3. Experimental Cloud Coefficient tables (see `fix/CloudCoeff`) in binary and netcdf4 formats
  4. Added 81 regression and 4 unit tests, see `README.md`.
  5. Updated: CMAQ-based (v4.x) Aerosols and Radiance/AOD simulation
  6. Improved loop-level performance: up to 5 times native improvement by optimizing loops.
- Multiple bugfixes vs. v2.3.x (see release notes)
- **Multiple new sensors added:**
  - EON MW, Sentinel3a SLSTR, Meteosat-11 Seviri. GOES-17 ABI 81K fix, Metop-C: AVHRR, IASI, SMAP/SMOS, TEMPEST-D, MI-LCOMS.v2, JPSS-2 VIIRS, GEOKOMPSAT-2a AMI, Metop-SG-A1 MWS, FY4a-GIIRS

CRTM v2.4.0 (cont)

CRTM v2.4.0

- **Bug Fixes:**

1. Bug in `CRTM_CloudCover_Define.f90`, fixing `Intent(in)` to `Intent(in out)` error for using gfortran compiler.
2. Bug in `CRTM_CloudCover_Define.f90`, when using the "Maximum-Random" scheme to calculate Total Cloud Cover.
3. Fixed a bug in `Common_RTSolution.f90`, for calculating surface emissivity Jacobian.
4. Setting the `` `atmosphere%cloud_fraction=ZERO` `` now removes cloudy radiances, previously it would default to a fully cloudy radiance. Enables smooth behavior from 1 to 0.
5. `NESDIS_ATMS_SnowEM_Module.f90` fix for bad input brightness temperatures, fails gracefully
6. OpenMP, ONLY because OpenMP fixed a bug whereby `GeometryInfo` used settings from one profile in the next profile - separated pointer nullification for OpenMP support.
7. `CRTM_MW_Ice_SfcOptics.f90`, `CRTM_MW_Snow_SfcOptics.f90` modified to allows channel-subsetting of microwave radiances: ensures that the correct channels are passing to the NESDIS Ice and Snow emissivity subroutines when only a subset of channels is available.

CRTM v2.4.1 (final scalar version)

CRTM v2.4.1 (under development, expected July 2021)

https://github.com/JCSDA/crtm

- **New features + bugfixes vs. v2.4.0:**
  - openMP extended, now supports both channel and profile loops (D. Abdi)
  - Fix: Snow cover emissivity when bad/missing observation data present (M. Chen)
  - Updates to compiler-specific configuration files
  - Updated support for Aerosol Coefficient files: CMAQ, GOCART, NAAPS (C. Dang)
  - Implemented two new aerosol coefficient look-up tables based on GOCART-GEOS5 and NAAPS aerosol specifications*(New aerosol species: nitrate and smoke)* (C. Dang)
  - Binary files / netCDF files hosted via git-LFS, gzipped for storage / speed.
  - Test codes updated to reflect/test new coefficient files

- **Sensors added:**
  - Corrections to internal sensor naming issues of abi_g17-81K
  - IASI-NG (testing), TROPICS_sv1_srf_v1, OMS GEMS-1/2 (in progress)
  - v2 ACCoeff and SpcCoeff for Metop-C / AMSU-A / MHS
  - ATMS-NG (in progress)
  - GOES-T ABI (testing/STAR)

CRTM v3.0 Goals and Work Plans

- **Cloudy Radiance** (B. Johnson)
  - 🟨 Backscattering coefficients for CRTM active sensor capability (Moradi, Johnson, Stegmann)
  - 🟩 Produce (Polarized) CRTM Scattering Coefficients
  - 🟩 Start systematic investigation of “optimal” single-scattering properties for CRTM applications
  - 🟩 Update of CHYM for microphysical consistency with NWP (B. Johnson, G. Thompson, Y. Lu, E. Clothiaux)
- **Surface** (M. Chen, Y. Zhu)
  - 🟩 Test CRTM-CSEM in GFS/GSI, focusing on the comparisons among model options.
  - 🟩 Analyze and document the tests of CRTM-CSEM in GFS/GSI.
  - 🟨 Initial implementation of MW ocean surface BRDF model.
  - 🟩 Ocean Surface Emissivity improvements IR (IRSSE, N. Nalli)
- **Full Polarization Solver Capability** (Q. Liu, T. Greenwald, B. Johnson, C. Cao)
  - 🟩 UV capable solver + polarization support under evaluation (CRTM v3.0-beta)
- **SW / IR improvements in CRTM**
  - 🟥 Cloud, surface, and aerosol impacts on visible channels C. Dang)
- **Aerosols update** (Johnson, Stegmann, S. Lu, M. Pagowski, B. Scherllin-Pirscher, others).
  - 🟩 Improved aerosol indices of refraction (via D. Turner, J. Gasteiger, C. Dang)
  - 🟩 Update of CRTM using initial CMAQ specifications (C. Dang, Y. Ma)
  - 🟩 Improve Lidar backscattering and attenuation calculations (Pagowski, Scherllin-Pirscher)

CRTM v3.0 Solver

μ , ϕ = cosine and azimuth of the viewing zenith angle (resp.)
μ_0 , ϕ_0= cosine and azimuth of solar zenith angle (respectively)
ϖ = single scattering albedo; τ = optical depth; $B(T)$ the Planck function at T; F_0= solar irradiance; P = phase function; and I = radiance.
$+\mu$ and $-\mu$ represent upward and downward directions

The vectorized radiative transfer equation in the case of a macroscopically isotropic and mirror-symmetric medium:

$$\mu \frac{d\boldsymbol{I}(\tau,\mu,\phi)}{d\tau} = \boldsymbol{I}(\tau,\mu,\phi) - \frac{\varpi}{4\pi}\int_0^{2\pi}\int_{-1}^{1} \times \boldsymbol{P}(\tau,\mu,\phi,\ \mu',\phi')\boldsymbol{I}(\tau,\ \mu',\phi')\mu' d\mu' d\phi' - \boldsymbol{S}(\tau,\mu,\phi,-\mu_0,\phi_0)$$

where

$$\boldsymbol{S}(\tau,\mu,\phi,\mu_0,\phi_0) = (1-\varpi)B(T)\begin{bmatrix}1\\0\\0\\0\end{bmatrix} + \frac{\varpi F_0}{4\pi}\begin{bmatrix}P_{11}(\tau,\mu,\phi,-\mu_0,\phi_0)\\P_{12}(\tau,\mu,\phi,-\mu_0,\phi_0)\\P_{13}(\tau,\mu,\phi,-\mu_0,\phi_0)\\P_{14}(\tau,\mu,\phi,-\mu_0,\phi_0)\end{bmatrix}e^{-\frac{\lambda\tau}{\mu_0}}$$

and

$$\boldsymbol{P} = \boldsymbol{L}(\pi - \boldsymbol{i}_2)\begin{bmatrix}F_{11}(\Theta) & F_{12}(\Theta) & 0 & 0\\ F_{12}(\Theta) & F_{22}(\Theta) & 0 & 0\\ 0 & 0 & F_{33}(\Theta) & F_{43}(\Theta)\\ 0 & 0 & -F_{43}(\Theta) & F_{44}(\Theta)\end{bmatrix}\boldsymbol{L}(-\boldsymbol{i}_1)$$

The vectorized radiative transfer equation is the same as the scalar radiative transfer except using the Stokes vector with 4 components: radiance (I), polarization difference (Q), the plane of polarization (U) and the ellipticity (V) of the electromagnetic wave.

Following Spurr 2006 (VLIDORT), the reciprocity principle can also be valid (in this case) for vectorized radiative transfer, so that transmission and reflection matrices from the top and from the bottom of a homogeneous layer can be the same.

P is the phase matrix (a product between the scattering matrix ***F***() and rotation matrices ***L)***.
The rotation angle ($\boldsymbol{i}_1$: incoming) ($\boldsymbol{i}_2$: outgoing) is the angle between scattering plane and the meridional plane.

Liu, Q. and Cao, C., 2019. Analytic expressions of the Transmission, Reflection, and source function for the community radiative transfer model. *Journal of Quantitative Spectroscopy and Radiative Transfer*, *226*, pp.115-126.

CRTM v3.0 Solver

$$\mathbf{P}(\tau,\mu,\phi,\ \mu',\phi') = \sum_{m=0}^{N} \frac{P_m^c(\tau,\mu,\mu')}{1+\delta_{0m}} \cos\left(m(\phi'-\phi)\right)$$
$$+P_m^s(\tau,\mu,\mu')\sin\left(m(\phi'-\phi)\right)$$

$$\mathbf{I}(\tau,\mu,\phi) = \sum_{m=0}^{N} I_m^c(\tau,\mu)\cos\ (m(\phi_0-\phi))$$
$$+I_m^s(\tau,\mu)\sin\ (m(\phi_0-\phi))$$

$$\mathbf{S}(\tau,\mu,\phi) = \sum_{m=0}^{N} S_m^c(\tau,\mu)\cos\ (m(\phi_0-\phi))$$
$$+S_m^s(\tau,\mu)\sin\ (m(\phi_0-\phi))$$

Liu, Q. and Cao, C., 2019. Analytic expressions of the Transmission, Reflection, and source function for the community radiative transfer model. *Journal of Quantitative Spectroscopy and Radiative Transfer*, *226*, pp.115-126.

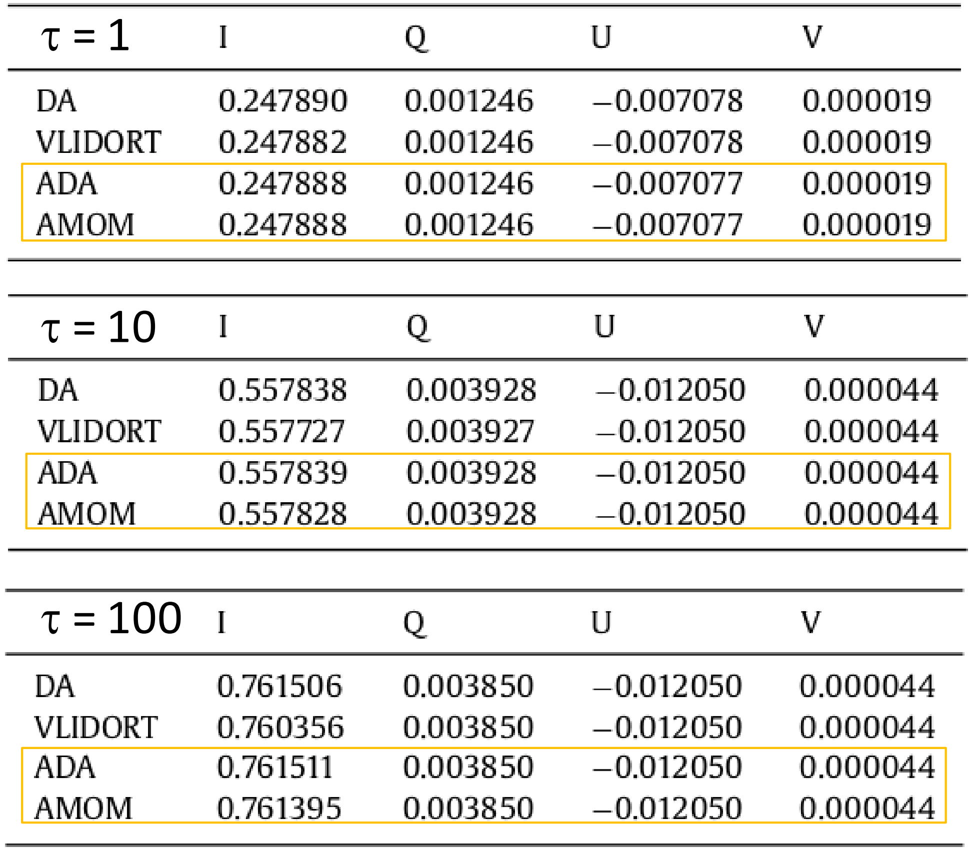
Since the first two components of the Stokes vector are even functions and the last two components of the Stokes vector are odd functions, The phase matrix, the source function, and the Stokes vector may be expanded as a series of cosine and sine functions.

$$\mu_i\begin{bmatrix} \frac{d\mathbf{I_m}(\tau,\mu_i)}{d\tau} \\ \frac{-d\mathbf{I}'_{\mathbf{m}}(\tau,-\mu_i)}{d\tau} \end{bmatrix} = \begin{bmatrix} \mathbf{I_m}(\tau,\mu_i) \\ \mathbf{I}'_{\mathbf{m}}(\tau,-\mu_i) \end{bmatrix}$$
$$-\varpi\sum_{j=1}^{N}\begin{bmatrix} \mathbf{P_m}(\tau,\mu_i,\mu_j) & \mathbf{P}'_{\mathbf{m}}(\tau,\mu_i,-\mu_j) \\ \mathbf{P}'_{\mathbf{m}}(\tau,\mu_i,-\mu_j) & \mathbf{P_m}(\tau,\mu_i,\mu_j) \end{bmatrix}$$
$$\times\begin{bmatrix} \mathbf{I_m}(\tau,\mu_j) \\ \mathbf{I}'_{\mathbf{m}}(\tau,-\mu_j) \end{bmatrix} w_j - \begin{bmatrix} \mathbf{S_m}(\tau,\mu_i,-\mu_0) \\ \mathbf{S}'_{\mathbf{m}}(\tau,-\mu_i,-\mu_0) \end{bmatrix}$$

The integration over μ for each Fourier component can be replaced by a discrete sum using Gaussian quadrature points (μ_i) and corresponding weights (w_i).

Following this, we extend the adding doubling algorithm (ADA) used in the CRTM solver SOI, and analytic matrix operator method (AMOM) to vectorized radiative transfer solvers.
See: `ADA_Module.f90` (v3.0 codebase)

CRTM v3.0 Solver

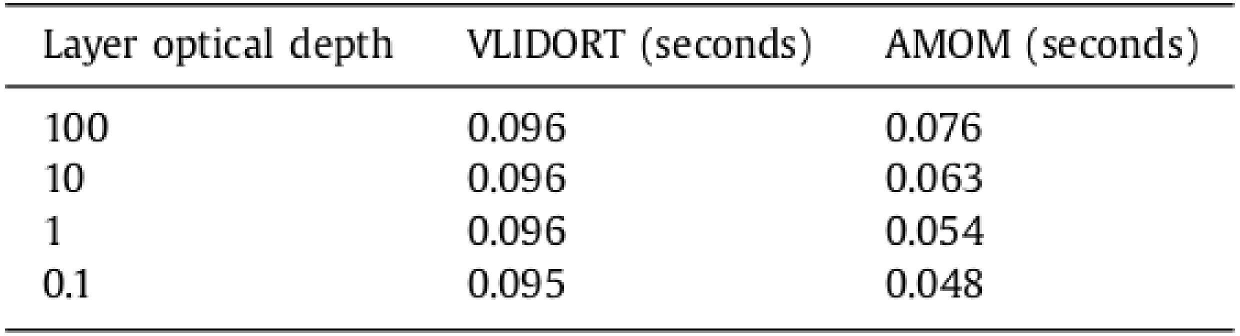


τ = 1	I	Q	U	V
DA	0.247890	0.001246	−0.007078	0.000019
VLIDORT	0.247882	0.001246	−0.007078	0.000019
ADA	0.247888	0.001246	−0.007077	0.000019
AMOM	0.247888	0.001246	−0.007077	0.000019

τ = 10	I	Q	U	V
DA	0.557838	0.003928	−0.012050	0.000044
VLIDORT	0.557727	0.003927	−0.012050	0.000044
ADA	0.557839	0.003928	−0.012050	0.000044
AMOM	0.557828	0.003928	−0.012050	0.000044

τ = 100	I	Q	U	V
DA	0.761506	0.003850	−0.012050	0.000044
VLIDORT	0.760356	0.003850	−0.012050	0.000044
ADA	0.761511	0.003850	−0.012050	0.000044
AMOM	0.761395	0.003850	−0.012050	0.000044

Surface albedo is 0.25. Single scattering albedo is 1.0. Results are 16 streams in a Rayleigh atmosphere of randomly oriented oblate spheroids.



Layer optical depth	VLIDORT (seconds)	AMOM (seconds)
100	0.096	0.076
10	0.096	0.063
1	0.096	0.054
0.1	0.095	0.048

Comparison of CPU time usage between VLIDORT and AMOM. Phase function for an atmosphere of randomly oriented oblate spheroids and 16 streams are used. The solar flux is normalized toπ, the solar zenith angle is 36.8699 (the cosine of the solar zenith angle is 0.8), and the surface albedo is 0.25. Single scat-tering albedo is 1.0. VLIDORT requires the single scattering is less than 1.0 and we use the single scattering albedo of 0.99999. The upwelling radiance is for a viewing (zenith) angle of 50.21°.

Liu, Q. and Cao, C., 2019. Analytic expressions of the Transmission, Reflection, and source function for the community radiative transfer model. *Journal of Quantitative Spectroscopy and Radiative Transfer*, *226*, pp.115-126.

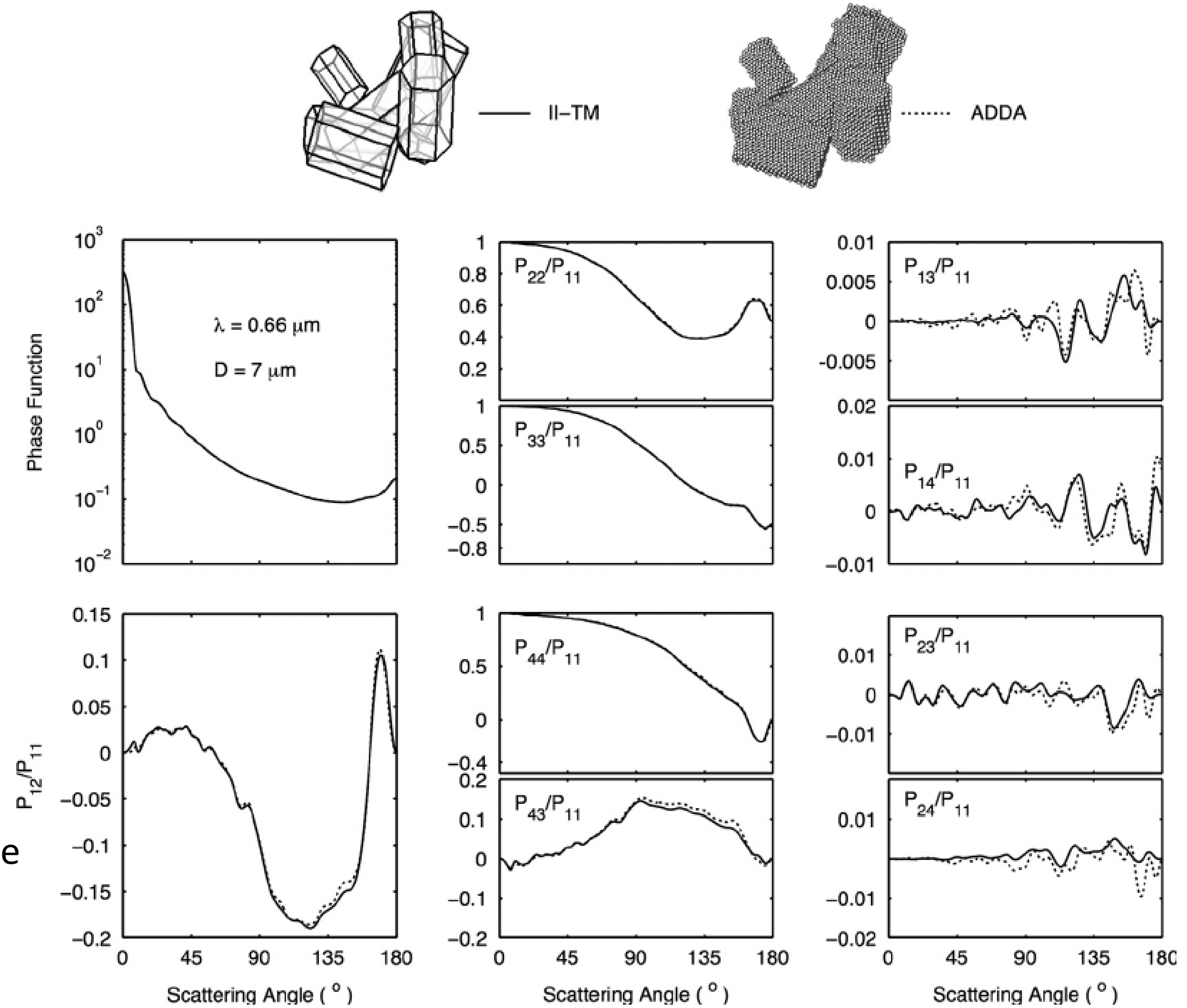
CRTM v3.0 Beta Focus

Cloud-impacted radiance and physical model simulation improvements (UV, VIS, IR, MW)

- Community Hydrometeor Model (CHYM)
  - Development continuing, and creating new polarized MW, IR, and VIS integrated cloud and aerosol scattering tables that are more closely linked with model assumed microphysical properties.
  - netCDF transition and conversion.
  - Updating Space-based radar support with linear polarization capabilities

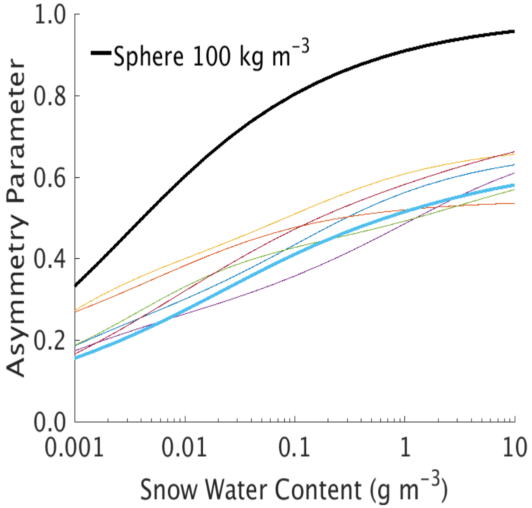
- *Coefficient tool development e.g.:*
  - https://github.com/PStegmann/INSPECT_CloudCoeff
  - https://github.com/JCSDA/CRTM_coef

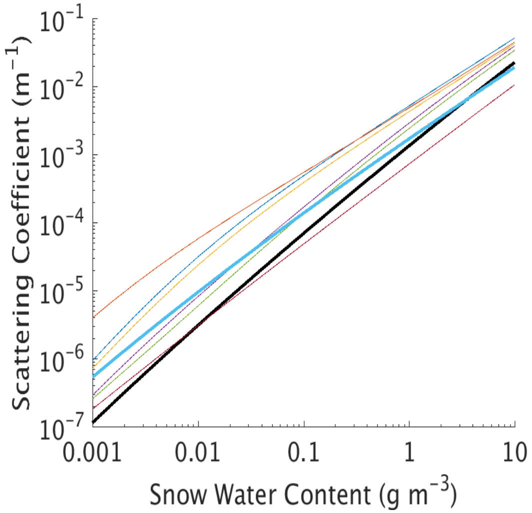
- **Community Surface Emissivity Model** (Ming Chen)
  - Extension of CSEM capabilities to support fully polarized surface BRDFs for ocean and land

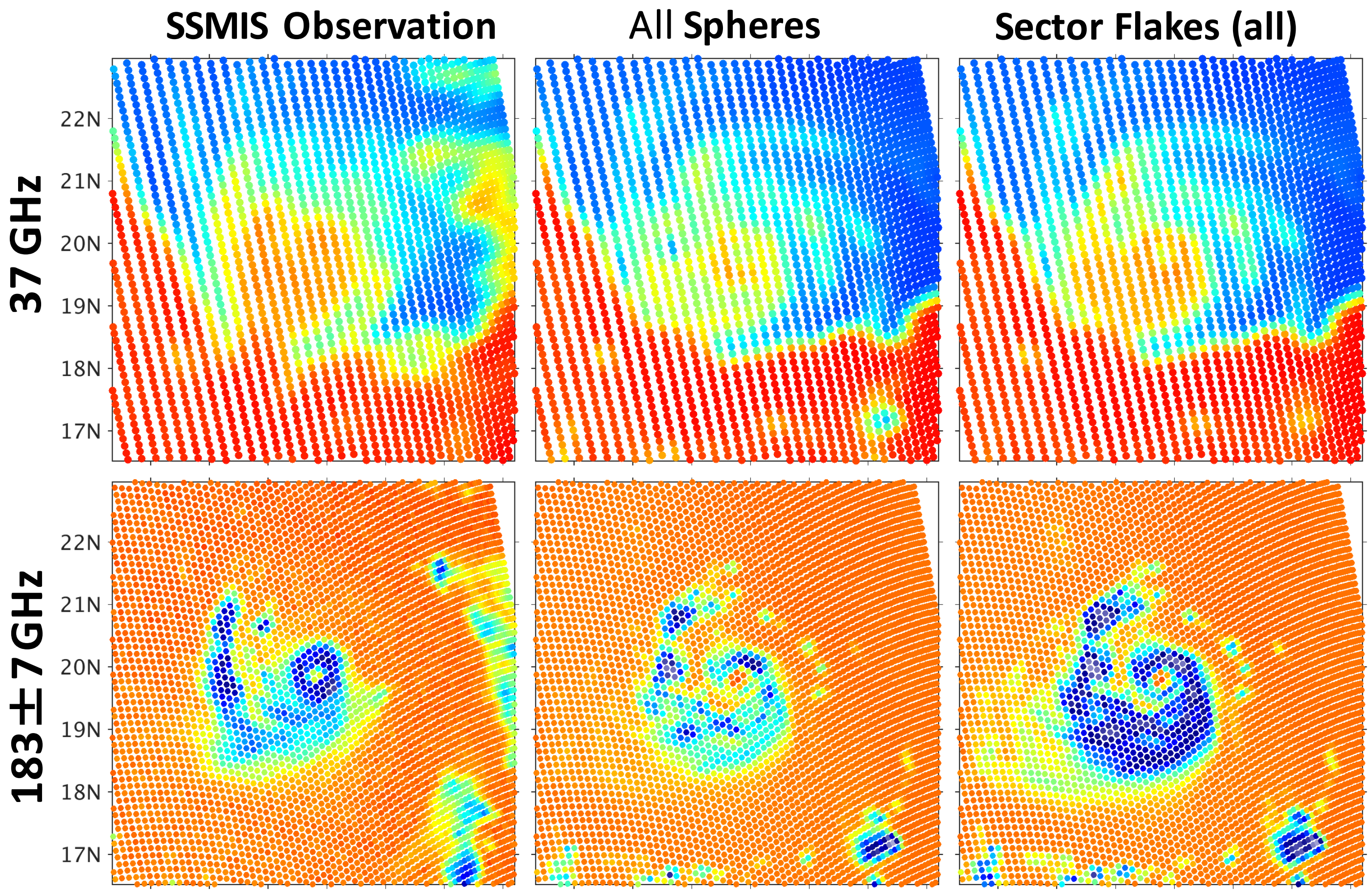


Build new LUT – improved scattering data

- Using sphere scattering table leads to systematic errors
  - Too much scattering at low frequencies
  - Too little scattering at high frequencies



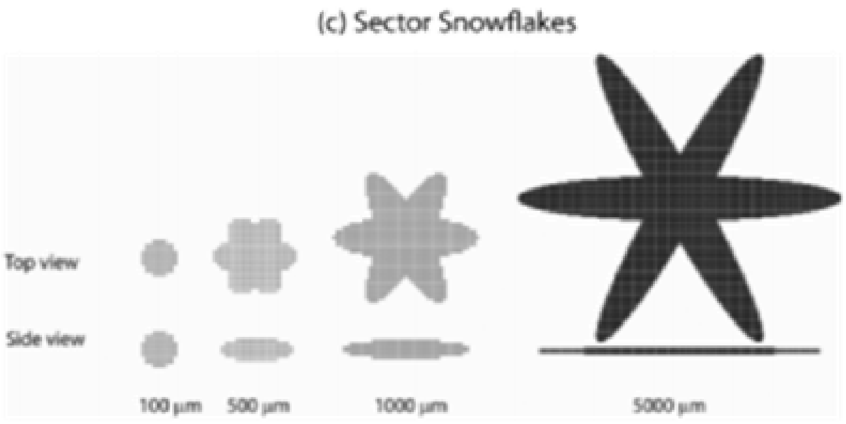


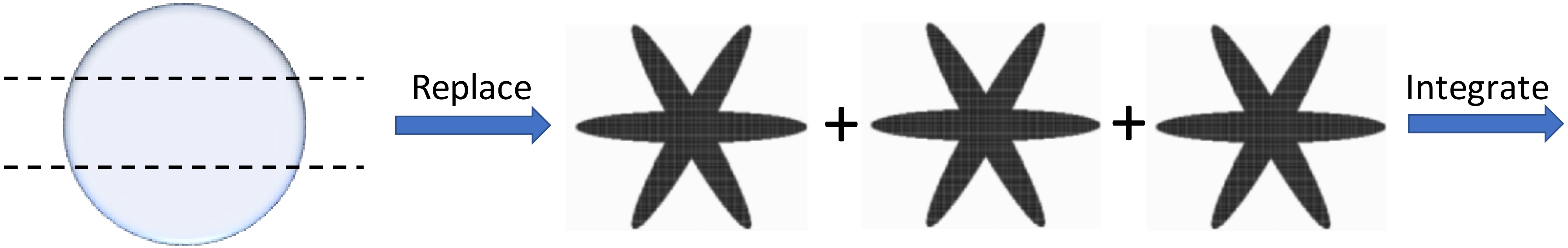


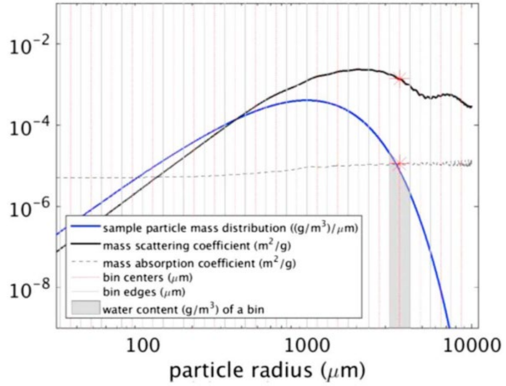
Slides courtesy of Yinghui Lu, Penn State

Build new LUT – improved scattering data

- Use scattering properties of sector snowflakes in Liu (2008) scattering database
- Replace sphere snow particles in model with the Liu (2008) sector snowflakes
  - One sphere snow is replaced by multiple sector snowflakes of the same size to preserve total mass of particles within each size bin
- New LUTs generated by integrating scattering properties over PSDs.







Slides courtesy of Yinghui Lu, Penn State

Commonly used PSD

- Mono-disperse (cloud ice/water)
- Gamma $N(D) = N_0 D^{\mu} e^{-\lambda D}$
- Exponential $N(D) = N_0 e^{-\lambda D}$
- Mixed (e.g., snow in Thompson 08 $N(D) = \frac{\mathcal{M}_2^4}{\mathcal{M}_3^3}\left[\kappa_0 e^{-\frac{\mathcal{M}_2}{\mathcal{M}_3}\Lambda_0 D} + \kappa_1\left(\frac{\mathcal{M}_2}{\mathcal{M}_3} D\right)^{\mu_s} e^{-\frac{\mathcal{M}_2}{\mathcal{M}_3}\Lambda_1 D}\right]$

	Rain	Snow	Graupel
GFDL MP	Exp ($N_0=8\times10^6$)	Exp ($\rho=100$, $N_0=3\times10^6$)	Exp ($\rho=400$, $N_0=4\times10^6$)
Thompson08	Exp (two-moment)	Exp + Gamma	Exp ($\rho=500$, $N_0=200/q_g$)
WSM6	Exp ($N_0=8\times10^6$)	Exp [$\rho=100$, $N_0=f(T)$]	Exp ($\rho=500$, $N_0=4\times10^6$)
And more …			

Slides courtesy of Yinghui Lu, Penn State

Per mess scattering properties

- Bulk mass and scattering coefficients

$$\beta_s = \int_0^\infty \sigma_s(D)N(D)dD \qquad m = \int_0^\infty m(D)N(D)dD$$

- Gamma distribution

$$N(D) = N_0 D^\mu e^{-\lambda D}$$

- CRTM LUTs use Scattering properties per mass.

$$\frac{\beta_s = \int_0^\infty \sigma_s(D)\cancel{N_0}D^\mu e^{-\lambda D}dD}{m = \int_0^\infty m(D)\cancel{N_0}D^\mu e^{-\lambda D}dD}$$

  - N_0 cancels!
  - Usually μ is fixed
  - Parameters λ determine scattering prop.

Slides courtesy of Yinghui Lu, Penn State

Calculate λ from model outputs

- Assume m-D relationship and Gamma distribution

$$m(D) = a\,D^b \qquad N(D) = N_0 D^\mu e^{-\lambda D}$$

- Single moment: output q (ρ_a can be calculated)

$$\rho_a q = IWC = \int_0^\infty a D^b N_0 D^\mu e^{-\lambda D}\,dD = aN_0 \frac{\Gamma(\mu+b+1)}{\lambda^{\mu+b+1}} \Longrightarrow \lambda = \left(\frac{aN_0\Gamma(\mu+b+1)}{\rho_a q}\right)^{\frac{1}{\mu+b+1}}$$

- Double moment: output q and N_t

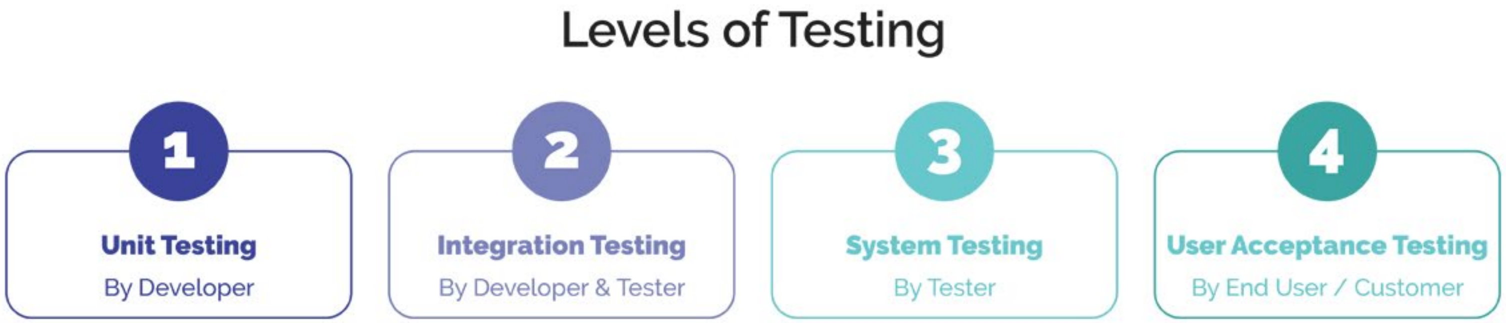
$$N_t = \int_0^\infty N_0 D^\mu e^{-\lambda D}\,dD = N_0 \frac{\Gamma(\mu+1)}{\lambda^{\mu+1}} \Longrightarrow \frac{\rho_a q}{N_t} = \frac{a\Gamma(\mu+b+1)}{\lambda^b\Gamma(\mu+1)} \Longrightarrow \lambda = \left(\frac{aN_t\Gamma(\mu+b+1)}{\rho_a q\Gamma(\mu+1)}\right)^{\frac{1}{b}}$$

Slides courtesy of Yinghui Lu, Penn State

CRTM v2.4.1 OpenMP (Abdi)

	Threads			Time	Speedup
	Total	Profiles	Channels		
Profile+Channels parallelization	1	1	1	0.85999	1.00x
	2	2	1	0.46468	1.85x
	4	4	1	0.27753	3.10x
	8	4	2	0.17958	4.80x
	12	**4**	**3**	0.12782	**6.73x**
Only channels parallelization (new)	1	1	1	0.88532	1.00x
	2	1	2	0.50481	1.75x
	4	1	4	0.30775	2.88x
	8	1	8	0.28052	3.16x
	12	**1**	**12**	0.25875	**3.42x**
Only profiles parallelization (v2.4)	1	1	1	0.8564	1.00x
	2	1	2	0.46808	1.83x
	4	**1**	**4**	0.28086	**3.05x**
Summary	Profile + channels		**6.73x**		AMD-12 Core CPU
	Channels only		**3.42x**		4-profiles & 12 channels
	Profiles only		**3.05x**		

Testing Framework



Levels of Testing

1 **Unit Testing**
By Developer

2 **Integration Testing**
By Developer & Tester

3 **System Testing**
By Tester

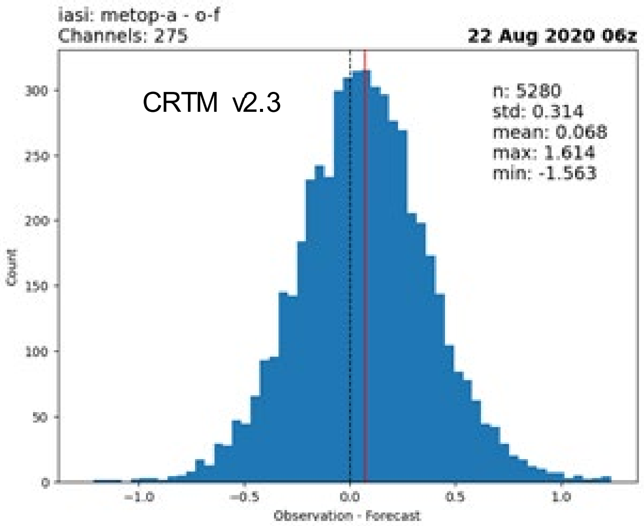
4 **User Acceptance Testing**
By End User / Customer

```
     Start 1: get_crtm_coeffs
 1/92 Test #1: get_crtm_coeffs  ... Passed 39.84 sec
     Start 2: test_check_crtm
 2/92 Test #2: test_check_crtm  ... Passed  3.30 sec
     Start 3: test_check_crtm_random
 3/92 Test #3: test_random      ... Passed 25.27 sec
     Start 4: Unit_TL_TEST
 4/92 Test #4: Unit_TL_TEST     ... Passed  0.10 sec
     Start 5: Unit_test_spc_io
 5/92 Test #5: Unit_test_spc_io ... Passed  0.09 sec
<...>
    Start 91: test_tangent_linear_ClearSky_v.abi_gr
91/92 Test #91: test_tangent_linear_ClearSky_v.abi_gr
...   Passed   0.18 sec
    Start 92: test_tangent_linear_ClearSky_modis_aqua
92/92 Test #92: test_tangent_linear_ClearSky_modis_aqua
...   Passed   0.10 sec

0 tests failed out of 92
```

```
.
├── application
├── regression
│   ├── adjoint
│   │   ├── test_ClearSky
│   │   └── test_Simple
│   ├── forward
│   │   ├── test_Aircraft
│   │   ├── test_AOD
│   │   ├── test_ChannelSubset
│   │   ├── test_ClearSky
│   │   ├── test_ScatteringSwitch
│   │   ├── test_Simple
│   │   │   └── coefficients
│   │   ├── test_SOI
│   │   ├── test_SSU
│   │   ├── test_VerticalCoordinates
│   │   └── test_Zeeman
│   ├── incsrc
│   ├── k_matrix
│   │   ├── test_AOD
│   │   ├── test_ChannelSubset
│   │   ├── test_ClearSky
│   │   ├── test_ScatteringSwitch
│   │   ├── test_Simple
│   │   ├── test_SOI
│   │   ├── test_SSU
│   │   ├── test_VerticalCoordinates
│   │   └── test_Zeeman
│   └── tangent_linear
│       ├── test_ClearSky
│       └── test_Simple
└── unit
    ├── input_output
    │   └── test_SpcCoeff
    └── Unit_Test
```

CRTM v2.4.0 Testing / Evaluation (H. Liu)

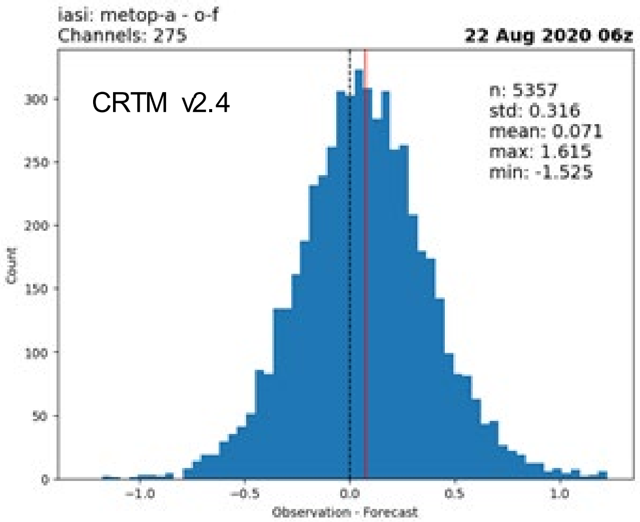


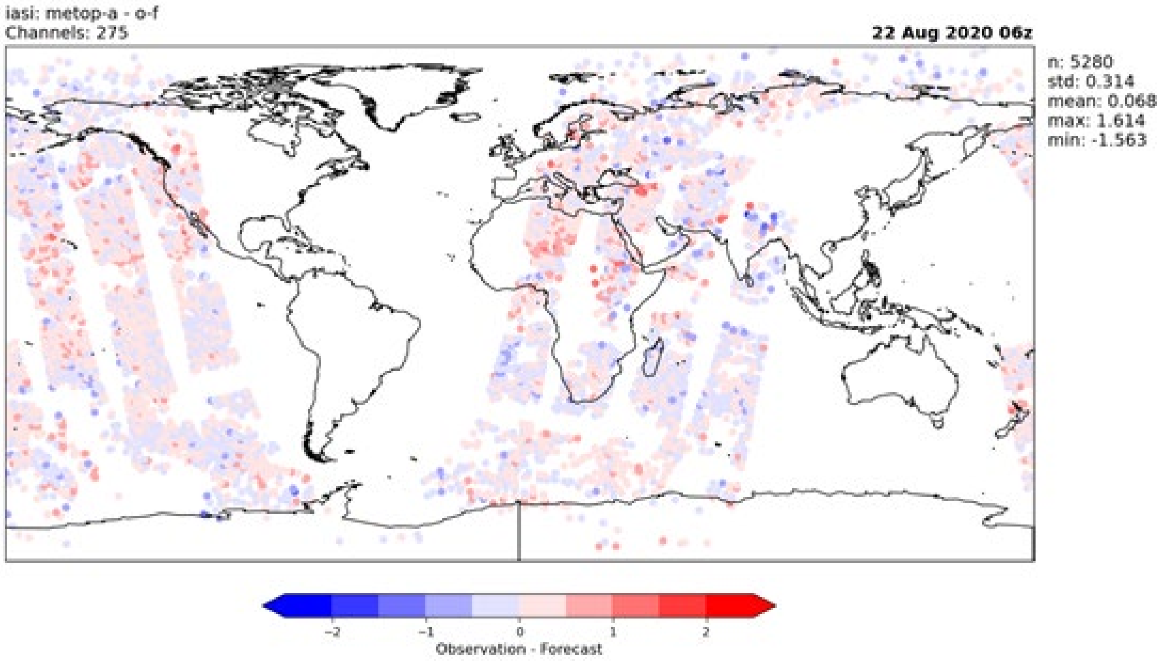
IASI

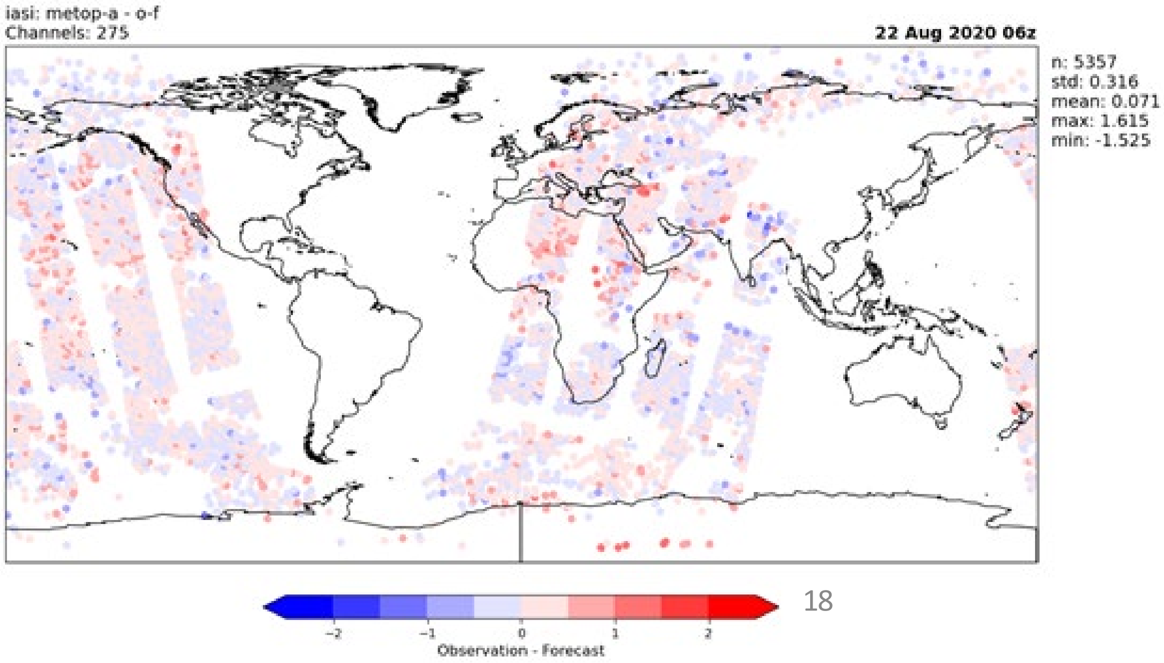
Nobs:

5280 v2.3.0

5357 v2.4.0



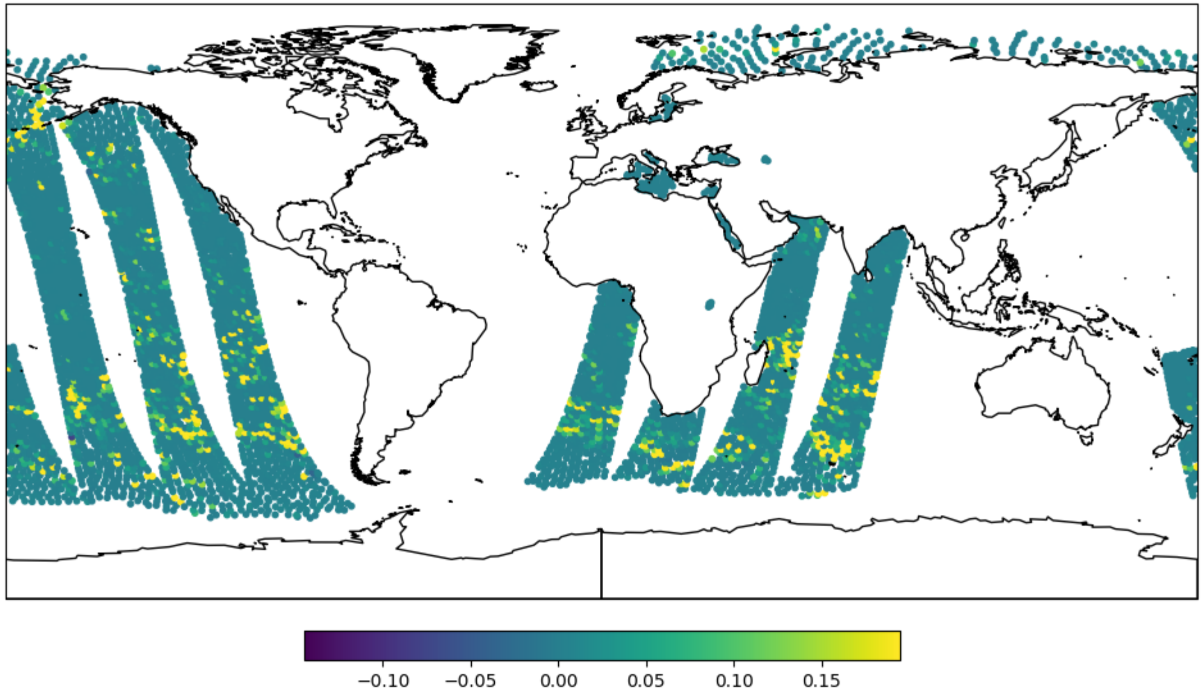


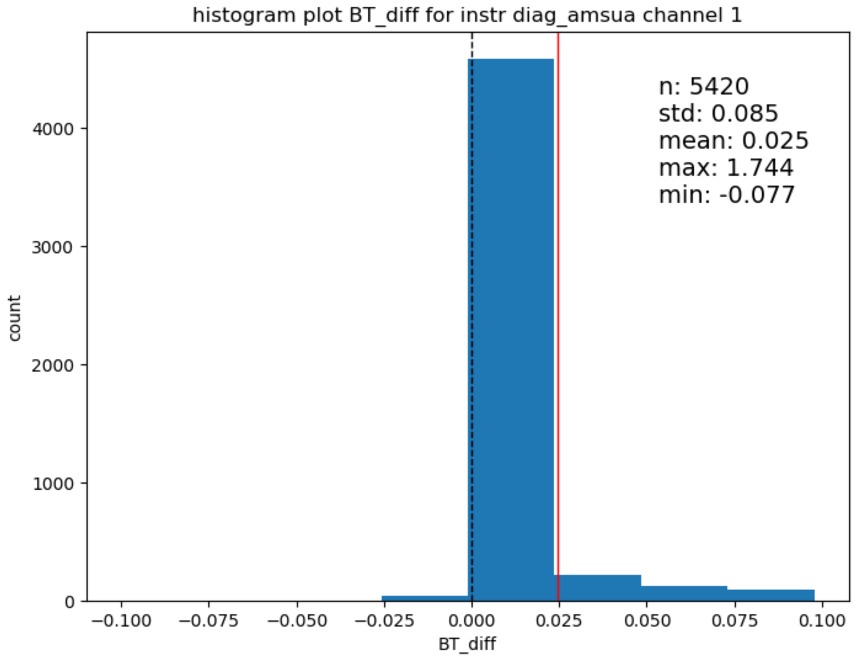


CRTM v2.4.0 Testing / Evaluation (H. Liu, M. Chen)

AMSU-A BT simulations show differences over sea (all-sky simulation), no differences over land, ice, snow, or mixed (clear-sky simulation).

BT Simulation diff v230-v240





Community Surface Emissivity Model (M. Chen)

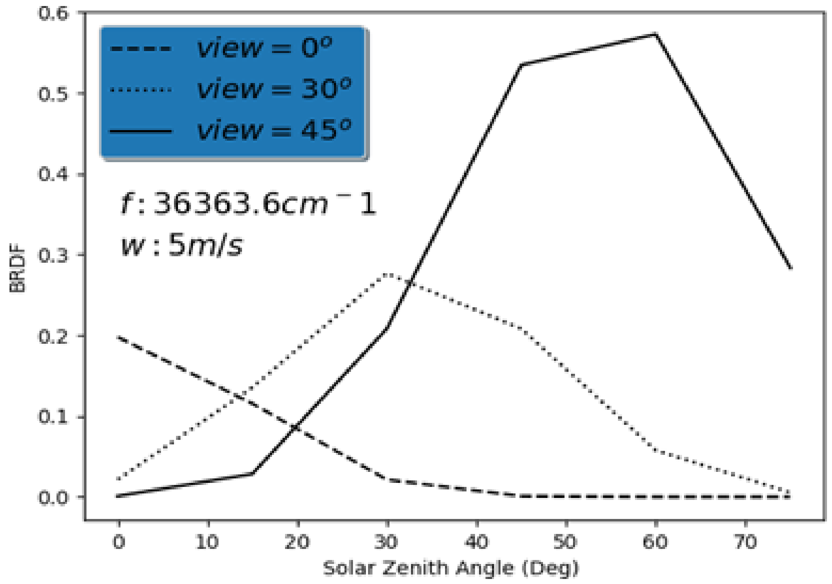
Updates on CSEM Development and Improvement

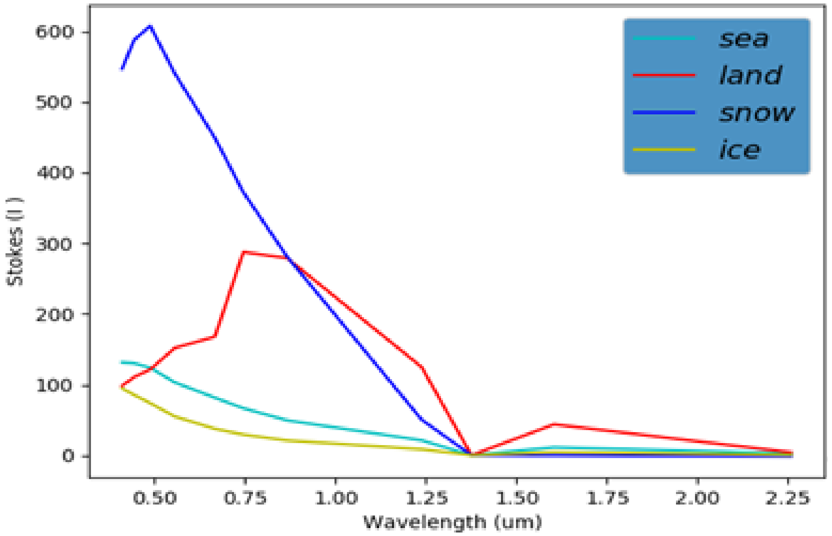
- The Visible-UV ocean surface BRDF model from Mark Liu’s CRTM 3.0beta package has been implemented into CSEM to support the CRTM 3.0 beta release. This model is similar to the IR ocean surface BRDF model, but with the extended water refractive index from 2.9um to 0.27um.

- A new Visible-UV surface reflectivity model was developed by using the NPOESS type-based spectral LUT to replace the constant value in the CRTM 3.0beta.

- The updated IR ocean surface emissivity LUTs from Nick Nalli have been implemented into CSEM. The CSEM package with these LUTs was delivered and is currently used to address the bias zonal dependency in GSI.

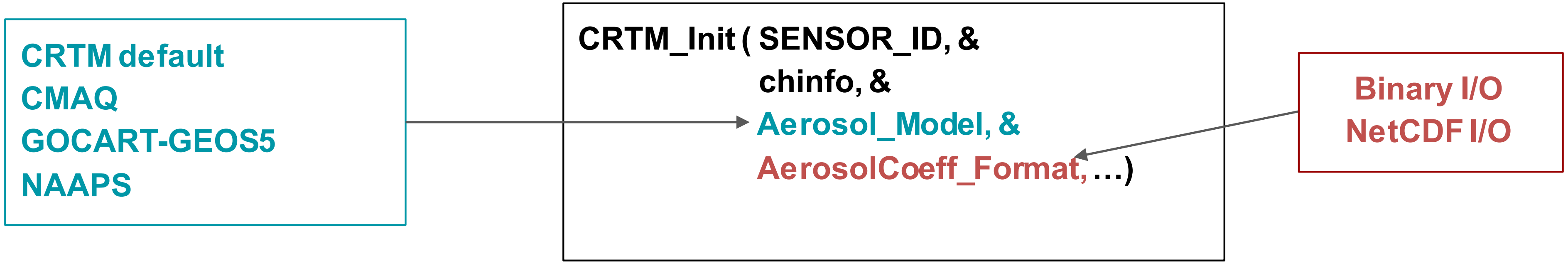
- The ATMS snow emissivity model has been improved to handle the invalid model inputs.





Updated Aerosol properties (C. Dang)

Overview of CRTM aerosol updates



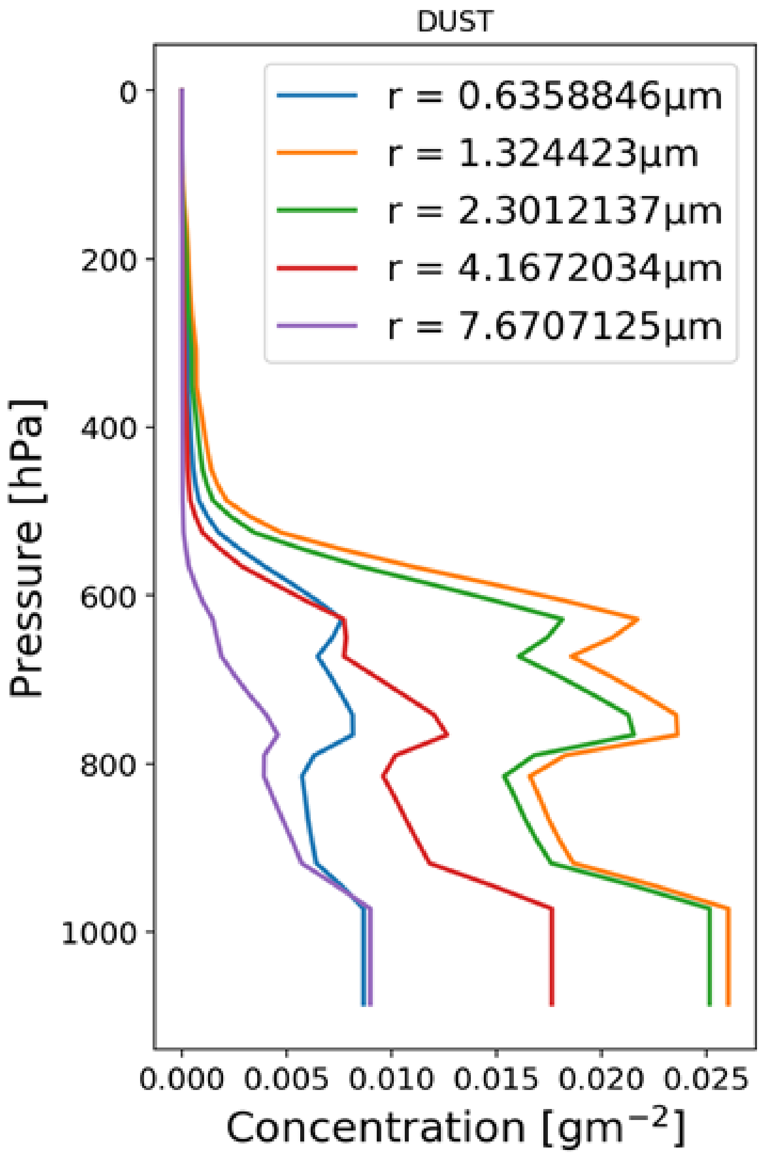
CRTM default
CMAQ
GOCART-GEOS5
NAAPS

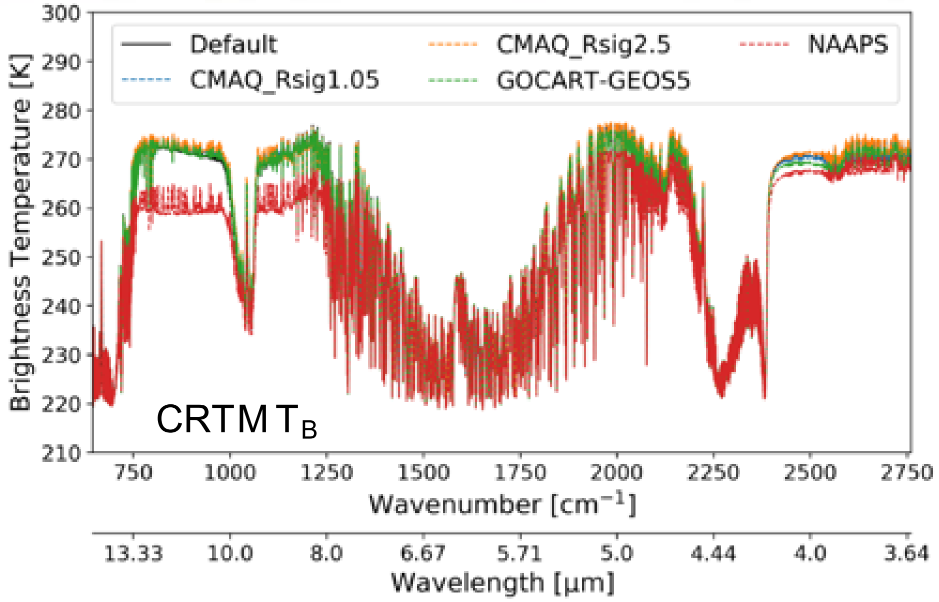
```
CRTM_Init ( SENSOR_ID, &
            chinfo, &
            Aerosol_Model, &
            AerosolCoeff_Format, ...)
```

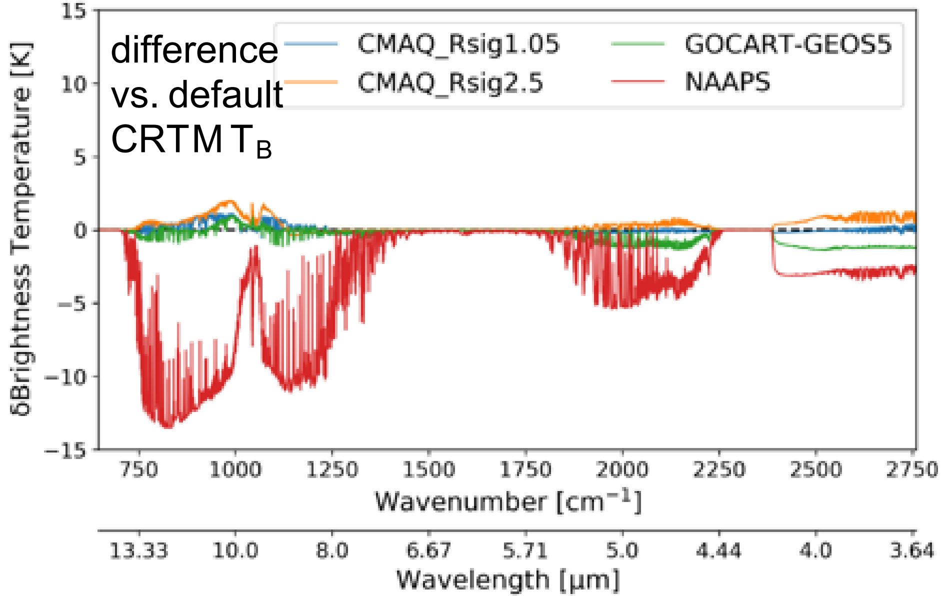
Binary I/O
NetCDF I/O

- Implemented two new aerosol coefficient look-up tables based on GOCART-GEOS5 and NAAPS aerosol specifications *(New aerosol species: nitrate and smoke).*
- Performed more tests on CRTM-AOD simulations with different aerosol schemes.
- Supported CMAQ AOD data assimilation.
- Next: develop aerosol coefficient generation packages.
- Next: evaluate CRTM simulations with different aerosol schemes.

Updated Aerosol properties (C. Dang)

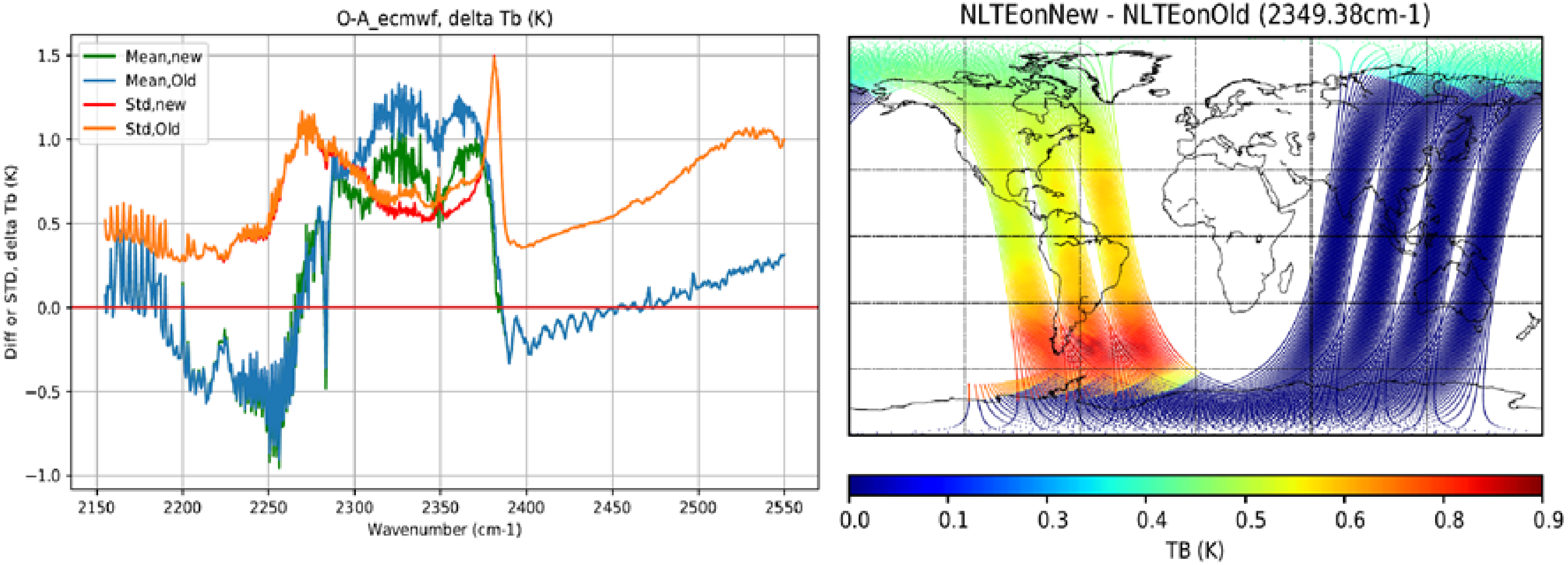






NLTE (Y. Ma)

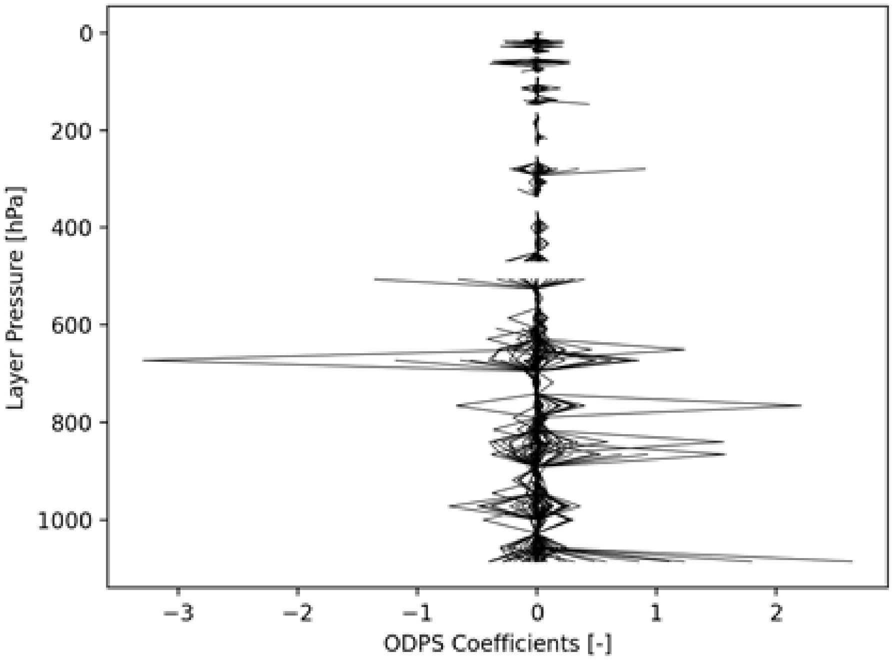
- Updated the Non-LTE correction coefficients for CrIS instruments on NOAA-NPP and NOAA_N20 based on the newly available vibrational temperature profiles for CO2 in 4.3 um band from M. López-Puertas et al. *(Milestone# CRTM3.3.3)*
- Worked with Haixia from EMC on assessing the impact of temperature induced SRF change with ABI-G17. *(Milestone# CRTM3.3)*



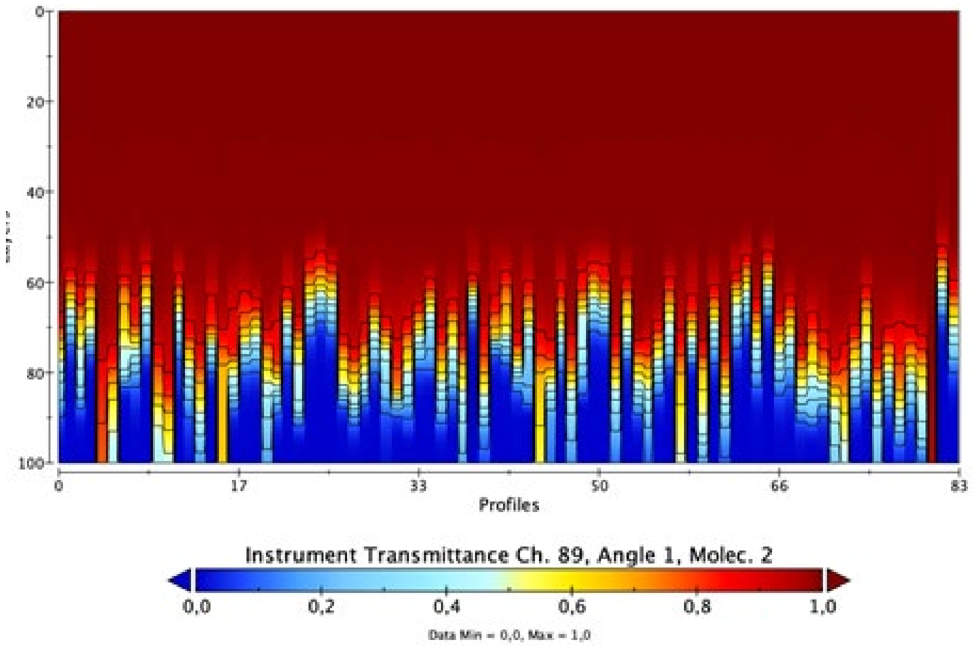
Left: Mean bias and standard deviation of O-A over the mid-latitude ocean after non-LTE correction. **Right:** Spatial distribution of the difference between the old and the new corrections.

Spectral expansion of CRTM radiance and sensor simulation capabilities (P. Stegmann)

IASI-NG coefficients created, under evaluation



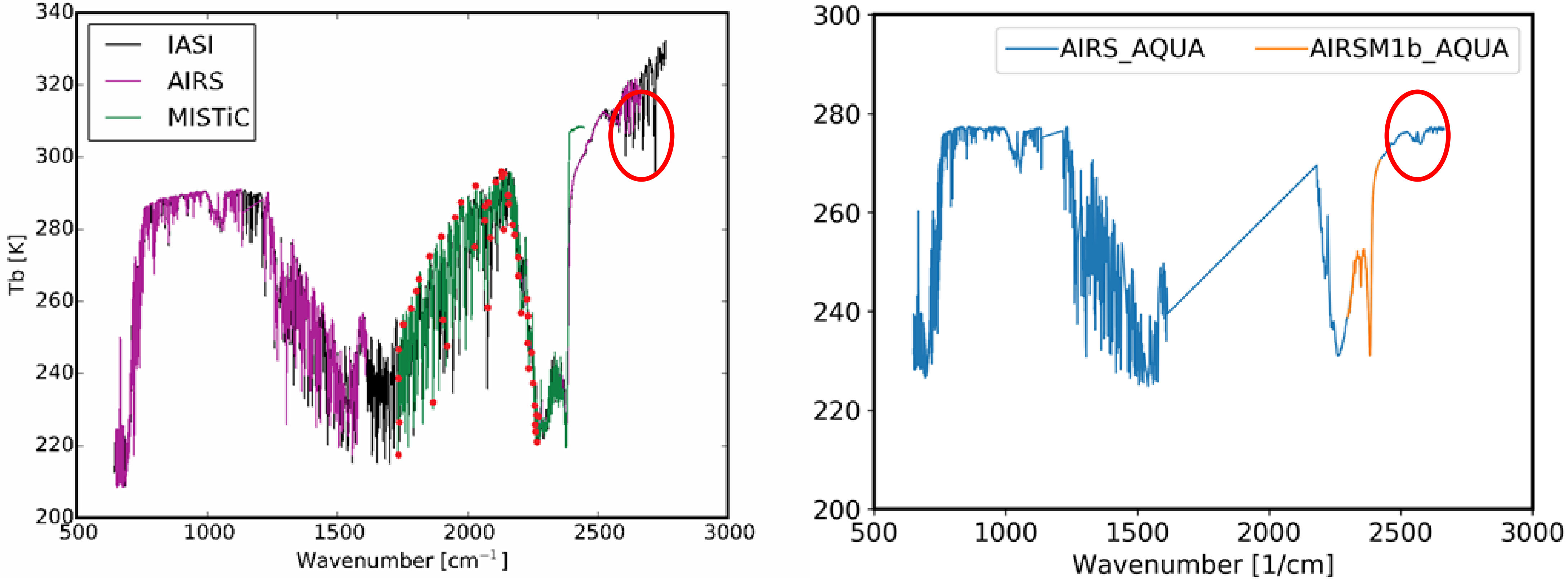
Vertical profile of ODPS coefficients for IASI-NG



Layer-to-space transmittance for Ch 89, IASI-NG
for each training profile from the ECMWF-83 set

Spectral expansion of CRTM radiance and sensor simulation capabilities (I. Moradi)

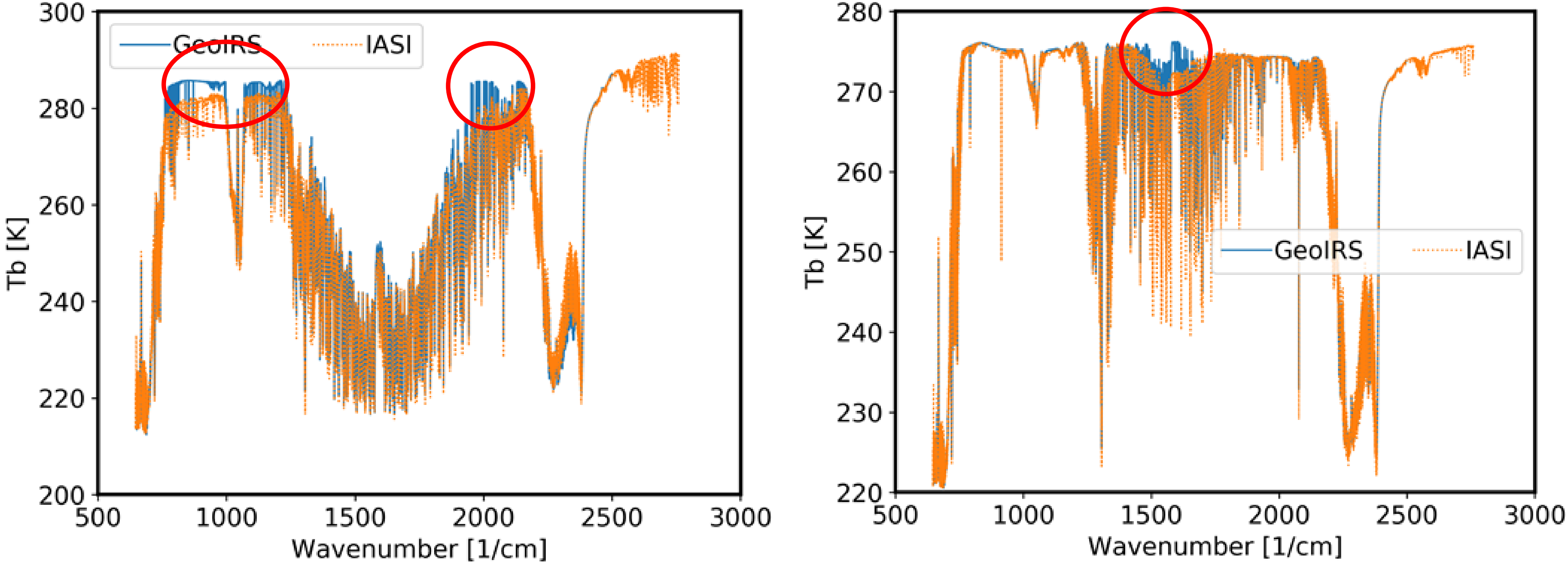
Generating CRTM Coefficients Hyperspectral Infrared Instruments



Some discrepancies exist between coefficients generated using available CRTM coefficient generation package and previous operational coefficients. After an extensive evaluation, the problem is caused by some inconsistencies in the bash files that are used to process Tape5 files. Although this issues was solved for 2500 1/cm but later it was found that the problem still exist for the higher frequencies (next slide).

Spectral expansion of CRTM radiance and sensor simulation capabilities (I. Moradi)

Generating CRTM Coefficients Hyperspectral Infrared Instruments



The problem seems to depend on the profiles used to run CRTM. The left image was created using Era profiles as input and right image using FASCOD profiles as input. GeoIRS is an instrument similar to MSG IRS and IASI shows the operational coefficients.

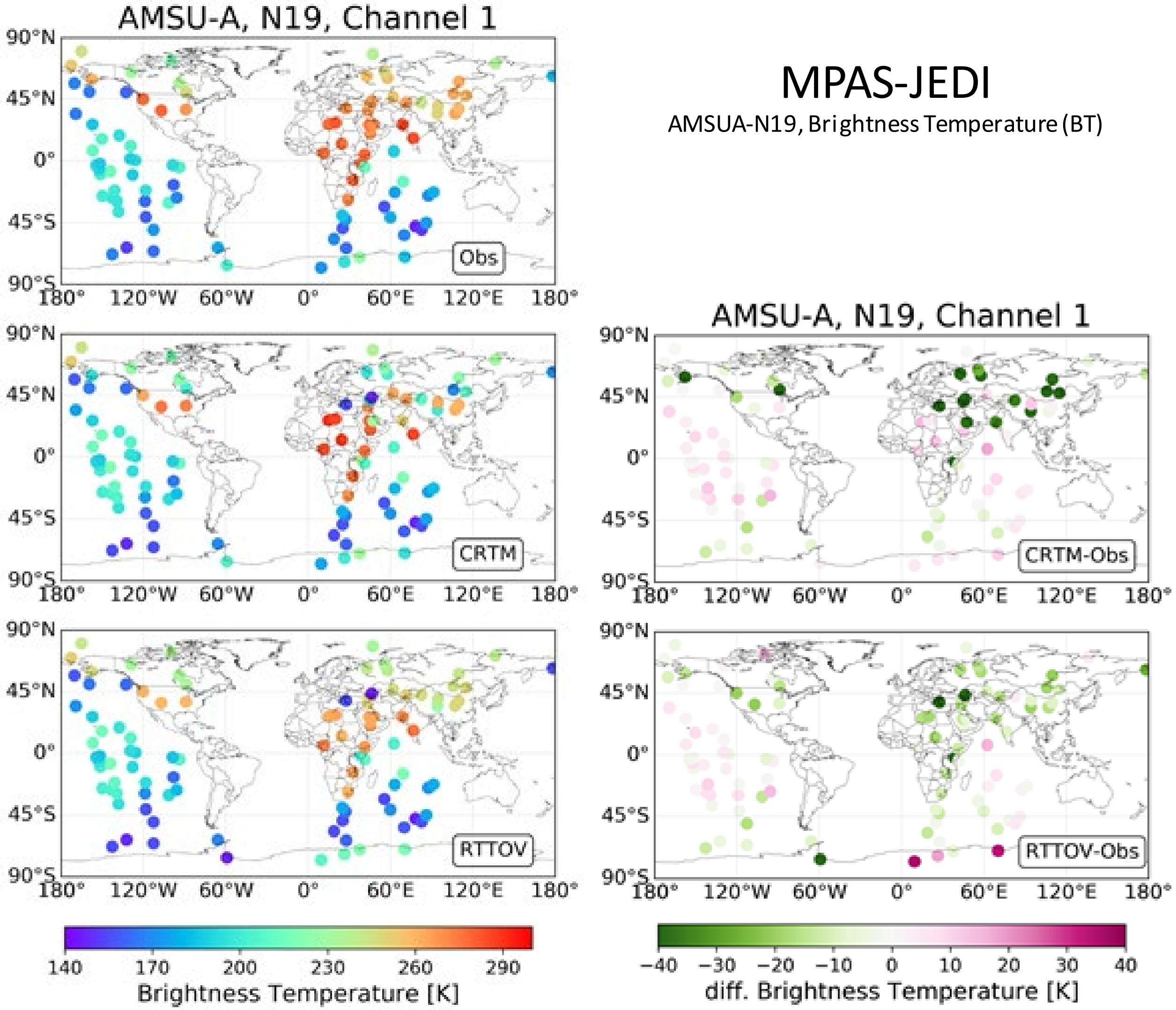
CRTM/RTTOV in JEDI

Current status:

- JEDI provides a flexible interface for users to select observational operators.
- The currently available operators are CRTM and RTTOV (with limited functions).
- Both operators are being developed and tested under fv3 and MPAS model background.
- More tests and evaluations are desirable.

MPAS-JEDI

AMSUA-N19, Brightness Temperature (BT)



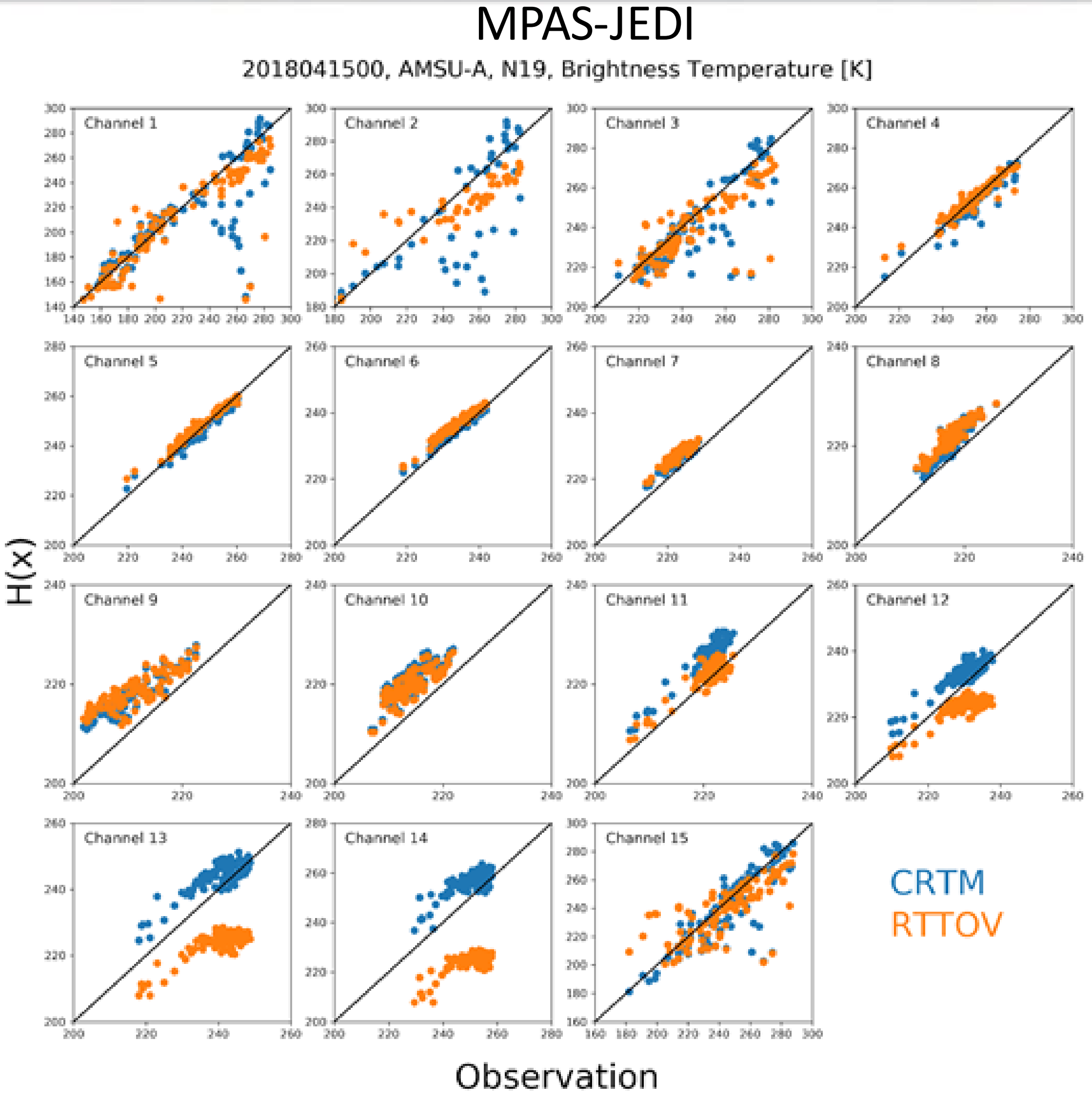
Figures: Brightness temperature (AMSUA Channel 1, 20180415) simulated using CRTM and RTTOV (clear sky case) with a test in MPAS-JEDI package. The left panels show observed BT, and BT simulated with CRTM and RTTOV. The right panels show the difference between H(x) and observations.

(Courtesy of Dr. Zhiquan Jake Liu for instructions on MPAS-JEDI).

CRTM/RTTOV in JEDI

JEDI as a platform for RT developments:

- As RT modelers, JEDI gives us easy access to a comprehensive observational and modeling dataset, based on which we can learn more about the strength and weaknesses of each RT model.
- For example, given the same inputs, both CRTM and RTTOV show consistent bias for channels 7-10; while for some channels, a model outperforms the other. (due to RT algorithms or assumptions?)
- *Future tests:*
  - Within the JEDI framework, evaluate CRTM/RTTOV for all channels against observations.
  - Cloudy-sky and aerosol-included cases.
  - If theoretically possible, will models be better constrained with both operators?



Figures: Comparison of BT simulated by CRTM and RTTOV vs. Observation for 15 AMSU-A channels. The spatial pattern of BT (channel 1) are shown in the previous slide.

CRTM 5-Year Plan Overview

1. **CRTM Core Development**
    - CRTM 3.x: Full polarization 1D fast RT [2021+], AI elements [2022+], 2D RT [2023+]
    - CRTM 4.x: C++/OO rewrite, generic, full-pol 3D fast RT, AI elements. [2025+]
    - Solver, Interface, Optimization [2021+]
2. **CRTM Application Development**
    - SIMOBS Application [2021+]
    - CRTM Support Applications: coefficients, AI elements, Python [2021+]
3. **Scientific and Technical Development**
    - Science support Aerosols, Clouds, Precipitation, Surface, Upward looking, Slant-path -> 3D, Transmittance, NLTE/Zeeman, improved Rayleigh scattering, etc. [2021+]
    - Support for spectral expansion (UV, subMM,FarIR): surface, clouds, aerosols, precipitation

Support / Contact

Website: https://www.jcsda.org/jcsda-project-community-radiative-transfer-model

Support:
https://groups.google.com/forum/#!forum/crtm-support

Support email:
crtm-support@googlegroups.com

Email: Benjamin.T.Johnson@noaa.gov for direct support, questions, and comments