Visiting Scientist mission report

Document NWPSAF-MO_VS-060

Version 1.0

17 April 2023

Comparison of RTTOV and CRTM in JEDI

Benjamin T. Johnson[1], James Hocking[2], David Rundle[2]

1 – Joint Center for Satellite Data Assimilation, University Corporation for Atmospheric Research, College Park, Maryland, USA
2 – The MetOffice, Exeter, UK

Comparison of RTTOV and CRTM in JEDI

Benjamin T. Johnson[1], James Hocking[2], David Rundle[2]

1 – Joint Center for Satellite Data Assimilation, University Corporation for Atmospheric Research, College Park, Maryland, USA
2 – The MetOffice, Exeter, UK

This documentation was developed within the context of the EUMETSAT Satellite Application Facility on Numerical Weather Prediction (NWP SAF), under the Cooperation Agreement dated 7 September 2021, between EUMETSAT and the Met Office, UK, by one or more partners within the NWP SAF. The partners in the NWP SAF are the Met Office, ECMWF, DWD and Météo France.

Change record			
Version	Date	Author / changed by	Remarks
0.1	25/10	Benjamin T. Johnson	Draft Version
0.2	23/11	Benjamin T. Johnson	Additional Details
0.9	27/11	Benjamin T. Johnson	First Release, missing results section
1.0	17.04.23	Benjamin T. Johnson	Final version

Comparison of RTTOV and CRTM in JEDI

Benjamin T. Johnson[1], James Hocking[2], and David Rundle[2]

[1]Joint Center for Satellite Data Assimilation, University Corporation for Atmospheric Research, College Park, Maryland, USA
[2]The Met Office, Exeter, UK

Abstract. This report summarises the work conducted during the visiting scientist mission NWPSAF_VS-020; accounts are given for the period 31.08.2022-22.09.2022. The objective of this mission was to provide a consistent methodology for intercomparing the Radiative Transfer for TOVs (RTTOV) and Community Radiative Transfer Model (CRTM), within the Joint Effort for Data Assimilation Initiative (JEDI) Unified Forward Operator (UFO) framework, developed at the Joint Center for Satellite Data Assimilation (JCSDA). An initial implementation of the RTTOV v12 model by co-authors Rundle and Hocking was used as a starting point for this work. A generic radiance operator interface was developed by co-author Johnson (visiting scientist) within the UFO, enabling CRTM, RTTOV, or both simultaneously to be used within the UFO, producing radiances based on the same geophysical parameters ("geovals"). Preliminary results of the intercomparison are presented herein.

1 Introduction

The Radiative Transfer for TIROS Operational Vertical Sounder model (RTTOV, e.g., Saunders et al., 2018) is able to simulate cloud and precipitation-affected radiances from microwave sensors (RTTOV-SCATT, Bauer et al., 2006) with sufficient accuracy (e.g., Barlakas et al., 2022) that they can be used for all-sky assimilation in weather forecasting (Geer et al., 2017) and for validating and improving forecast models of cloud and precipitation (Forbes et al., 2016). Version 12 represents the latest operational version of RTTOV, and this is the version used in the present intercomparison study.

The Joint Center for Satellite Data Assimilation (JCSDA) Community Radiative Transfer Model (CRTM, e.g., Han et al., 2006) is a fast, 1-D radiative transfer model used in numerical weather prediction, calibration / validation, etc. across multiple federal agencies and universities. It simulates all-sky, all-surface radiances in the visible, infrared, submillimeter, and microwave regions of the electromagnetic spectrum on a per sensor basis. Like RTTOV, in addition to providing the forward-modeled radiances, CRTM also produces a Jacobian (i.e., the sensitivity of the radiance with respect to a change in the model state), enabling satellite data assimilation methodologies.

In the current UFO implementation, the interface to CRTM and RTTOV are specific to the requirements of each operator, requiring the user to modify / craft special YAML and provide RTTOV or CRTM specific input files. The present research performed at the MetOffice sought to develop a generic radiance operator, with the ultimate goal of disambiguating the specific requirements of CRTM or RTTOV when calling codes require simulated radiances. The primary capabilities were accomplished during this 3 week visit, which are described in this document.

2 Tools Used

The Joint Effort for Data assimilation Integration (JEDI) is a versatile data assimilation (DA) system for Earth System Prediction. The JEDI software package can be run on a variety of platforms from laptops to supercomputers, for a variety of purposes, from teaching and learning DA fundamentals to the development and validation of new DA algorithms and observational operators, to leading-edge atmospheric and oceanic research, to operational weather forecasting. It is designed to readily accommodate new atmospheric and oceanic models and new observation systems.

The function of the Unified Forward Operator (UFO) is to provide a unified system of forward operator developments and interfacing with internal geophysical properties coming from the JEDI model(s). The forward operator relates the physical properties to the observed radiances or other operators, such as radiosondes, aircraft measurements, and/or other categories of observations. In the data assimilation literature, this procedure is often represented by the expression $H(\mathbf{x})$. Here $\mathbf{x}$ represents the vector of prognostic variables on the model grid, typically obtained from a forecast or analysis; and H represents the observation operator" that generates simulated observations from that model state. The sophistication of observation operators varies widely, from in situ measurements where it may just involve interpolation and possibly a change of variables (e.g. radiosondes), to remote sensing measurements that require physical modeling to produce a meaningful result (e.g. radiance, GNSSRO).

The flexibility of UFO permits the execution of multiple forward operators simultaneously on the same geophysical values, including radiance operators. The primary advantage of this capability in the current context is to ensure that the input physical quantities are the same, and consequently, any variances observed between CRTM and RTTOV lie within the configuration or model assumptions.

RTTOV interface implementation and details

RTTOV (Radiative Transfer for TOVS) is a very fast radiative transfer model for passive visible, infrared and microwave downward-viewing satellite radiometers, spectrometers and interferometers. It is a FORTRAN 90 code for simulating satellite radiances, designed to be incorporated within user applications [Saunders, 2018]

In this study, we chose to use RTTOV v13, based on an existing implementation by David Rundle, and James Hocking. The interface with UFO was updated to utilize the "default" settings to provide an "apples-to-apples" comparison between RTTOV and CRTM, these are listed in Appendix B.

CRTM interface implementation and details

The CRTM version 2.4.0 was used for this study. Released in October, 2020, v2.4.0 is the current operational version of CRTM used in NOAA NWP frameworks. Substantial improvements over v2.3.0 make v2.4.0 a more appropriate choice for this intercomparison.
In particular, the `release/crtm_jedi_v2.4.0` branch of the CRTM repository at https://github.com/JCSDA-internal/crtm was used.

3 Methodology

Generic radiance operator development

The primary contribution of this work was to enable a capability to operate both CRTM and RTTOV simultaneously on the same data, producing radiance and jacobian outputs.

In order to accomplish this, a generic "Radiance" operator was developed, that is designed to be independent from the actual RT model being used. This is expected to have long term benefits of being able to add different radiance operators, while preserving the separation of concerns.

The following UFO files were created / modified to enable this capability:

Created: `ufo/src/ufo/operators/radiance/` which contains:

```
CMakeLists.txt
Crtm/
ObsRadiance.cc
ObsRadiance.h
ObsRadianceTLAD.cc
ObsRadianceTLAD.h
rttov/
rttovcrtm/
```

Where `crtm` and `rttov` are interfaced through the usual YAML parameters. The new parameter, `rttovcrtm` permits both radiance operators to run simultaneously. This directory contains the new files:

```
CMakeLists.txt
ObsRadianceRTTOVCRTM.cc
ObsRadianceRTTOVCRTM.h
ObsRadianceRTTOVCRTM.interface.F90
ObsRadianceRTTOVCRTM.interface.h
ObsRadianceRTTOVCRTMTLAD.cc
ObsRadianceRTTOVCRTMTLAD.h
ObsRadianceRTTOVCRTMTLAD.interface.F90
ObsRadianceRTTOVCRTMTLAD.interface.h
rttovcrtmParameters
ufo_radiancerttovcrtm_mod.F90
ufo_radiancerttovcrtm_utils_mod.F90
```

Some work remains to make this method fully generic. Ultimately, we wish to be able to specify "radiance" as the operator with an option to determine which radiance operators will be used, currently, the specific fast model must be specified within the YAML.

Testing setup

Once the code modifications were completed, a new testing setup was created. A simple forward operator test, based on existing tests, we extended to support the "rttovcrtm" option.

Two primary tests will be performed: (1) a stand-alone radiance operator test, and (2) the full data assimilation system test, using a hybrid 3DVAR with model inputs from GEOS and GFS and observations from AMSU-A, IASI, MHS, GPM GMI, and SEVIRI on various satellite platforms.

4 Conclusion and Future Work

The present work only includes the preliminary generic interface, which is not fully generic because the RT model selection (e.g., CRTM, RTTOV, or RTTOVCRTM) must be specified within the YAML file.

This work will continue until we completely disambiguate the generic Radiance operator from the specific choice of RT model. Specifically, it is anticipated that the code will attempt to find an available RT model (e.g., a module file or a repository within the bundle) and default to that during build configuration, freeing up the operational user from dealing with the RT model specific configurations.

Additionally, support for easily adding new RT models should be included, to enable ease of access for contributors. This would likely be a python script that would generate the necessary source code to conform to the input requirements of the model.

References

Saunders, R., Hocking, J., Turner, E., Rayer, P., Rundle, D., Brunel, P., Vidot, J., Roquet, P., Matricardi, M., Geer, A., Bormann, N., and Lupu, C., 2018: An update on the RTTOV fast radiative transfer model (currently at version 12), Geosci. Model Dev., 11, 2717-2737, https://doi.org/10.5194/gmd-11-2717-2018.

Han, Y., 2006: JCSDA Community Radiative Transfer Model (CRTM): version 1. NOAA Technical Report, 122, https://repository.library.noaa.gov/view/noaa/1157.

Bauer, P., Lopez, P., Benedetti, A., Salmond, D. and Moreau, E., 2006. Implementation of 1D+ 4D-Var assimilation of precipitation-affected microwave radiances at ECMWF. I: 1D-Var. Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography, 132(620), pp.2277-2306.

Barlakas, V., Galligani, V.S., Geer, A.J. and Eriksson, P., 2022. On the accuracy of RTTOV-SCATT for radiative transfer at all-sky microwave and submillimeter frequencies. Journal of Quantitative Spectroscopy and Radiative Transfer, 283, p.108137.

Appendix A. Acronyms and Definitions

1D-VAR	One Dimensional Variational (Data Assimilation)
ABI	Advanced Baseline Imager
AD	Adjoint model
ADA	Advanced Doubling Adding method
AER	Atmospheric and Environmental Research
AIRS	Atmospheric Infrared Sounder
AMSU	Advanced Microwave Sounding Unit
AOD	Aerosol Optical Depth
ARMS	Advanced Radiative Transfer Modeling System
CD	Continuous Development
CI	Continuous Integration
CMAQ	Community Multiscale Air Quality (model)
CRTM	Community Radiative Transfer Model
CrIS	Cross-track Infrared Sounder
DA	Data Assimilation

ECMWF	European Centre for Medium-range Weather Forecasts
EMC	Environmental Modeling Center (NOAA)
EPA	Environmental Protection Agency
FASTEM	Fast microwave Emissivity Model
GeoVal	Geophysical values at locations
GFDL	Geophysical Fluid Dynamics Laboratory
GOCART	Goddard Chemistry Aerosol Radiation and Transport
GSI	Gridpoint-Statistical Interpolation system
IASI	Infrared Atmospheric Sounding Interferometer
IFS	Integrated Forecast System (EMCWF)
IR	Infrared
IRSSE	Thermal IR Sea Surface Emissivity Model
JCSDA	Joint Center for Satellite Data Assimilation
JMA	Japan Meteorological Agency
JEDI	Joint Effort for Data Assimilation Integration
LBL	Line-by-line
LBLRTM	Line by Line Radiative Transfer Model
LTE	Local Thermodynamic Equilibrium
LUT	Lookup Table
MetOp	Meteorological Operational satellite
MW	Microwave
NAAPS	Navy Aerosol Analysis and Prediction System
NASA	National Aeronautics and Space Administration
netCDF	Network Common Data Form
NOAA	National Oceanic and Atmospheric Administration
non-LTE	Non Local Thermodynamic Equilibrium
NPOESS	National Polar-orbiting Operational Environmental Satellite System

NPP	(Suomi) National Polar-Orbiting Partnership satellite
NRL	Naval Research Laboratory
ODAS	Optical Depth in Absorber Space
ODPS	Optical Depth in Pressure Space
OpenMP	Open Multi-Processing
OPTRAN	Optical Path Transmittance (model)
PR	Pull Request
pyCRTM	Python CRTM
R2O	Research to Operations
RT	Radiative Transfer
RTTOV	Radiative Transfer for TOVS
SOI	Successive-Order-of-Interactions method
SRF	Spectral Response Function
SSMI	Special Sensor Microwave Imager
SSMI/S	SSMI / Sounder
SSU	Stratospheric Sounding Unit
TIROS	Television Infrared Observation Satellite
TL	Tangent-Linear model (also TLM)
TOVS	TIROS Operational Vertical Sounder
TROPICS	Time-Resolved Observations of Precipitation structure and storm Intensity with a Constellation of Smallsats
UFO	Unified Forward Operator
UV	Ultraviolet
VIS	Visible
WSM-6	WRF Single-Moment 6-Class Scheme
WRF	Weather Research & Forecasting Model

Appendix B. “Default” RTTOV v13 settings used in the present study:

```
General configuration

- fix_hgpl recommended true, apply_reg_limits better false for
intercomparisons

    self % rttov_opts % config % apply_reg_limits        = .false. !< Switch
to restrict input profiles to coef training limits
    self % rttov_opts % config % verbose                 = .true.  !< Switch
for verbose output
    self % rttov_opts % config % do_checkinput           = .true.  !< Switch
to apply internal profile checking
    self % rttov_opts % config % fix_hgpl                = .true.  !< Switch
to apply fix to match 2m p with elevation in geometry calculations


Interpolation

- interpolation is required, the different interpolation modes do impact
radiances by a few tenths of a Kelvin. Mode 4 is recommended when the number
of input levels exceeds the number of coefficient levels and mode 1 otherwise,
but we might use a mix of 54L and 101L coefficients with e.g. an 80L model.
Since we are using mode 4 operationally at the Met Office let's go with that.
- lgradp can be enabled if we care about pressure Jacobians, otherwise not
needed.
- spacetop and reg_limit_extrap should both be true

    self % rttov_opts % interpolation % addinterp        = .true.        !<
Switch to enable RTTOV interpolator
    self % rttov_opts % interpolation % interp_mode      = 4
        !< Interpolation mode (valid options 1-5, see user guide)
    self % rttov_opts % interpolation % lgradp           = .false.       !<
Switch to make pressure an active variable in TL/AD/K models
    self % rttov_opts % interpolation % spacetop         = .true.        !<
Switch to assume space boundary at top-most input pressure level
    self % rttov_opts % interpolation % reg_limit_extrap = .true.        !<
Switch to extrapolate input profiles using regression limits


General RT options - gases

- if we supply ozone profiles (would be nice), we should set ozone_data .true.
- we should try to make sure that assumptions about CO2 profile(s) are similar
between RTTOV and CRTM. Without an explicit profile, RTTOV's reference profile
has a max of ~405 ppmv. CRTM is likely to be different. We can use variable
O3+CO2 RTTOV coefficients and supply a scaled copy of the reference profile to
match the max concentration in CRTM (RTTOV provides a simple method to achieve
this), or supply (an) explicit profile(s) to RTTOV (and CRTM?). In either case
co2_data must be .true.
- use RTTOV reference/background for other gases (for the proposed
coefficients to be used these gases will not be allowed to vary anyway)

    self % rttov_opts % rt_all % ozone_data              = .true.        !<
Switch to enable input of O3 profile
    self % rttov_opts % rt_all % co2_data                = .true.        !<
Switch to enable input of CO2 profile
```

```
    self % rttov_opts % rt_all % n2o_data              = .false.       !<
Switch to enable input of N2O profile
    self % rttov_opts % rt_all % co_data               = .false.       !<
Switch to enable input of CO profile
    self % rttov_opts % rt_all % ch4_data              = .false.       !<
Switch to enable input of CH4 profile
    self % rttov_opts % rt_all % so2_data              = .false.       !<
Switch to enable input of SO2 profile


General RT options - other

- I've specified what I'd generally recommend.
- If CRTM doesn't make use of 2m T or q, then set use_t2m_opdep and/or use_q2m
to false. (Similarly can set to false if there are no explicit near-surface T
or q fields in the model data being used).
- when do_lambertian is false, RTTOV assumes specular surface reflection: for
simplicity let's stick with this for now.
- if comparing Jacobians, switchrad sets the units: when false Jacobians are
in terms of radiance, when true they are BT (for thermal channels only)

    self % rttov_opts % rt_all % addrefrac             = .true.   !< Switch
to enable atmospheric refraction
    self % rttov_opts % rt_all % switchrad             = .true.   !< Switch
for input units in AD/K models
    self % rttov_opts % rt_all % use_q2m               = .true.   !< Switch
to enable use of 2m q variable
    self % rttov_opts % rt_all % use_t2m_opdep         = .true.   !< Switch
to enable use of 2m T variable in opdep param (false to match v12.3)
    self % rttov_opts % rt_all % do_lambertian         = .false. !< Switch
for setting Lambertian reflection (IR and MW)
    self % rttov_opts % rt_all % lambertian_fixed_angle  = .true.   !< Switch
for fixed/parameterised effective angle for Lambertian option
    self % rttov_opts % rt_all % plane_parallel        = .false. !< Switch
to ignore atmospheric curvature
    self % rttov_opts % rt_all % rad_down_lin_tau      = .true.   !< Linear-
in-tau or layer-mean for downwelling radiances
    self % rttov_opts % rt_all % transmittances_only   = .false. !< Switch
to only compute transmittances, no radiances
    self % rttov_opts % rt_all % dtau_test             = .false. !< Switch
to apply dtau test in transmit/integrate calculations


MW RT options

- surface emissivity treatment TBD, but recommend FASTEM-6 over sea, no user
foam fraction input
- assume we will not include CLW absorption

    self % rttov_opts % rt_mw % fastem_version         = 6        !< FASTEM
version (0-6); 0 => TESSEM2
    self % rttov_opts % rt_mw % supply_foam_fraction   = .false. !< Supply a
foam fraction to FASTEM
    self % rttov_opts % rt_mw % fastem3_rwd_fix        = .true.   !< Switch
to enable the FASTEM-3 relative wind direction bug fix
```

```
    self % rttov_opts % rt_mw % clw_data                = .false. !< Switch
to enable input of cloud liquid water profile
    self % rttov_opts % rt_mw % clw_scheme              = 2       !< MW CLW
scheme: 1 => Liebe, 2 => Rosenkranz, 3 => TKC
    self % rttov_opts % rt_mw % clw_cloud_top           = 322     !< Lower
pressure limit for MW CLW calculations (hPa)


UV/VIS/IR RT options

- surface emissivity treatment TBD, but recommend IREMIS (option 2) over sea
- disable NLTE correction (for now at least!)
- let's avoid solar to begin, can look at that later. Solar BRDF model and
Rayleigh settings are as recommended.
- aerosols and clouds are disabled for now, can come back to this after clear-
sky is running

    self % rttov_opts % rt_ir % ir_sea_emis_model       = 2       !< IR sea
emissivity model (1-2)
    self % rttov_opts % rt_ir % do_nlte_correction      = .false. !< Switch
to enable NLTE bias correction

    self % rttov_opts % rt_ir % addsolar                = .false. !< Switch
to enable solar simulations
    self % rttov_opts % rt_ir % solar_sea_brdf_model    = 2       !< Solar
sea BRDF model (1-2)
    self % rttov_opts % rt_ir % rayleigh_single_scatt   = .true.  !< Switch
to enable Rayleigh single-scattering for VIS/NIR channels
    self % rttov_opts % rt_ir % rayleigh_max_wavelength  = 2._kind_real !<
Ignore Rayleigh scattering for channels at wavelengths above this (microns)
    self % rttov_opts % rt_ir % rayleigh_min_pressure   = 0._kind_real !<
Ignore Rayleigh scattering at pressures below this (hPa)

    self % rttov_opts % rt_ir % addaerosl               = .false. !< Switch
to enable IR aerosol calculations
    self % rttov_opts % rt_ir % user_aer_opt_param      = .false. !< Switch
to supply aerosol optical properties explicitly per channel
    self % rttov_opts % rt_ir % addclouds               = .false. !< Switch
to enable IR cloudy calculations
    self % rttov_opts % rt_ir % user_cld_opt_param      = .false. !< Switch
to supply cloud optical properties explicitly per channel

        self % rttov_opts % rt_ir % grid_box_avg_cloud       = .true.  !<
Switch to supply grid-box average cloud concentration or cloud

    self % rttov_opts % rt_ir % cldcol_threshold        = -one          !<
Ignore cloud columns with weights lower than this
    self % rttov_opts % rt_ir % cloud_overlap           = one           !<
Select cloud overlap 1=max/random, 2=simple (use with caution)
    self % rttov_opts % rt_ir % cc_low_cloud_top        = 750._kind_real !<
Upper pressure limit for simple overlap option (hPa)
    self % rttov_opts % rt_ir % ir_scatt_model          = ir_scatt_chou  !<
IR scattering model to use
    self % rttov_opts % rt_ir % vis_scatt_model         = vis_scatt_dom  !<
VIS/NIR scattering model to use
```

```
    self % rttov_opts % rt_ir % dom_nstreams            = 8              !<
Number of DOM streams, must be even and not less than 2
    self % rttov_opts % rt_ir % dom_accuracy            = 0._kind_real   !<
Convergence criterion for termination of DOM azimuthal loop
    self % rttov_opts % rt_ir % dom_opdep_threshold     = 0._kind_real   !<
DOM ignores levels below this optical depth:
    self % rttov_opts % rt_ir % dom_rayleigh            = .true.         !<
DOM includes Rayleigh scattering for VIS/NIR channels


RTTOV-SCATT (MW scattering) options

- again let's get clear-sky working first, but a few notes here:
- the totalice setting should probably be false, we will need to discuss
optical properties
- for all RTTOV options, use the same settings as above
- the exception is refraction: delta-Eddington is really a plane-parallel
solver, so it is probably best to be consistent (though the impact of
including refraction in the clear-sky calculations will be small compared to
other errors/differences)
- also ozone_data: only applies to ICI
- ice_polarisation: interesting to test this, but if we compare against
unpolarised CRTM this can be set to zero


      self % mw_scatt % use_totalice        = .false.        !< False =>
separate ice and snow; True => total ice
      self % mw_scatt % mmr_snowrain        = .true.         !< Snow and rain
input units are: False => kg/m2/s; True => kg/kg

      !Copy all corresponding RTTOV options to RTTOV_SCATT options structure
      self % mw_scatt % opts % config                      = self %
rttov_opts % config
      self % mw_scatt % opts % use_q2m                     = self %
rttov_opts % rt_all % use_q2m
      self % mw_scatt % opts % use_t2m_opdep               = self %
rttov_opts % rt_all % use_t2m_opdep
      self % mw_scatt % opts % fastem_version              = self %
rttov_opts % rt_mw % fastem_version
      self % mw_scatt % opts % fastem3_rwd_fix             = self %
rttov_opts % rt_mw % fastem3_rwd_fix
      self % mw_scatt % opts % supply_foam_fraction        = self %
rttov_opts % rt_mw % supply_foam_fraction
      self % mw_scatt % opts % interp_mode                 = self %
rttov_opts % interpolation % interp_mode
      self % mw_scatt % opts % reg_limit_extrap            = self %
rttov_opts % interpolation % reg_limit_extrap
      self % mw_scatt % opts % lgradp                      = self %
rttov_opts % interpolation % lgradp
      self % mw_scatt % opts % dtau_test                   = self %
rttov_opts % rt_all % dtau_test
      self % mw_scatt % opts % rad_down_lin_tau            = self %
rttov_opts % rt_all % rad_down_lin_tau

      self % mw_scatt % opts % ozone_data                  = .false.
```

```
      self % mw_scatt % opts % addrefrac                    = .false.

      self % mw_scatt % opts % ice_polarisation             = 1.4_jprb
      self % mw_scatt % opts % lusercfrac                   = .false.
!< Manually supply effective cloud fraction (default is false)
      self % mw_scatt % opts % cc_threshold                 = 0.001_kind_real
!< Threshold for determining if scattering calculations will be performed
(v12.3 default is 0.05 but that is very large)
      self % mw_scatt % opts % hydro_cfrac_tlad             = .true.
!< Switch for hydrometeor TL/AD sensitivity to effective cfrac (default is
true).
      self % mw_scatt % opts % zero_hydro_tlad              = .false.
!< Switch for hydrometeor TL/AD sensitivity in layers with zero hydrometeor
concentration (default is false).


PC models

- not interested in these for now so deactivate

    self % rttov_opts % rt_ir % pc % addpc               = .false. !< Switch
to enable PC-RTTOV
    self % rttov_opts % htfrtc_opts % htfrtc             = .false. !< Switch
to use htfrtc
```