RTTOV v14 User Interface Changes

James Hocking

Met Office, 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**	**Remarks**
0.1	29/02/24	JH	First draft for beta
1.0	03/09/24	JH	Changes post-beta
1.0.1	15/10/24	JH	Updates after Met Office review
1.0.2	28/11/24	JH	Updates after DRR

Contents

1. Introduction

The interface to RTTOV v14.0 has changed significantly compared to RTTOV v13.x. One of the major changes in v14.0 is bringing the capabilities of the RTTOV-SCATT model (for microwave scattering) inside RTTOV and thus enabling scattering across the full spectrum within RTTOV itself. The wider aim, to be developed further in future releases, is to eliminate spectral distinctions wherever possible to provide better spectral consistency. RTTOV v14.0 is the first step in this process. Since this already involves substantial updates, it was decided to make further user interface changes at the same time to improve clarity and consistency in the interface, and with the aim of having smaller interface updates with future major releases.

Most user interface changes are relatively superficial, and involve changes in variable, derived type, and subroutine names, and/or subroutine argument order. Some changes are more fundamental, primarily the way the atmospheric profile is represented, how cloud/hydrometeor scattering simulations are run, and the input/output of surface emissivity and reflectance.

This document provides a comprehensive description of the user interface differences between RTTOV v13.2 (and v13.1, v13.0 which are very similar) and v14.0 with the aim of helping users update their code. This document should also be helpful for users of RTTOV v12, v11 and even v10 who are updating to v14.

2. Summary of changes since v13.2

The most significant changes are the modifications to the input profile structure in how levels and layers are represented (see section 9), updates related to surface emissivity and reflectance (section 11), and the implementation of the RTTOV-SCATT capabilities within RTTOV (section 12). The latter means that scattering simulations for microwave sensors are now done in a very similar way to those for IR sensors from a technical perspective, with benefits including increased flexibility compared to RTTOV-SCATT, and generalisation of many features to sensors in all spectral regions. It also means that the entire RTTOV-SCATT interface (including associated derived types and subroutines) has been removed from the code.

The following list summarises the main new features and changes in RTTOV v14.0 since v13.2. The most significant changes from a technical implementation perspective are highlighted in **bold red**.

General:

- **Revise the input profile structure so that all input variables (temperature, gases, scattering inputs) are provided on the same vertical grid. This improves consistency with NWP model fields.**
- New option to output overcast BTs in addition to existing overcast radiances.
- New option to enable UV/VIS/NIR Jacobians in terms of reflectance (now the default) instead of radiance.
- New diagnostic output structure containing per-profile outputs, currently geometric heights of pressure half- and full-levels, and computed effective hydro fraction.
- Improved user-level routines for checking inputs to RTTOV before running full simulations.

- Improved notification to users via the *radiance%quality(:)* output when input values are clipped by various parameterisations within RTTOV.
- Zeeman coefficients based on v13 predictors enabled.
- **Default values of various options have changed since RTTOV v13.**

Surface emissivity/reflectance:

- **Emissivity and reflectance inputs/outputs refactored into a single new *rttov_emis_refl* data structure.**
- Enable full user control over diffuse reflectance at all wavelengths in the same way as for emissivity and BRDF.
- Diffuse reflectance is added to BRDF from sea sun-glint model for VIS channels to allow for consistency in treatment of ocean colour/sub-surface scattering.
- USGS water reflectance datasets extended to the UV (used for sea surface reflectance).
- **Enable fully flexible heterogenous surface capability: users can specify surface and near-surface properties for multiple surfaces per profile, and the properties are combined before the radiance solver is called.**
- Implement interface to CAMEL v3 IR land surface emissivity atlas datasets.
- Enable optional return of nearby land IR emissivity/BRDF within a user-specified distance if atlas has no data at the original location.
- Emissivity retrieval output structure generalised for dynamic emissivity retrievals in clear-sky cases, and for Chou-scaling solver in addition to delta-Eddington, and for all cloud overlap options.

Scattering:

- **Scattering for MW sensors now run through the main RTTOV interface in a very similar way to IR sensors from a technical perspective (the separate RTTOV-SCATT model no longer exists).**
- Delta-Eddington solver implemented within RTTOV for infrared and microwave sensors.
- Radar solver implemented in RTTOV, and passive radiances are computed alongside radar reflectivities.
- Cloud overlap options from RTTOV-SCATT available as additional options in RTTOV.
- Consistent unit conversions applied for hydrometeor concentrations in the UV/VIS/IR and MW.
- Allow separate units selection for hydrometeors and aerosols.
- UV/VIS/IR hydrometeor optical properties made fully flexible allowing any combination of particle types to be used in the same simulation (as implemented in earlier RTTOV versions for aerosols and MW hydrometeors).
- Explicit optical property inputs can be used for MW simulations as well as UV/VIS/IR.
- Explicit optical property phase functions and Legendre coefficients are no longer active variables in the TL/AD/K.
- Tang *et al* modification to Chou-scaling solver implemented to improve accuracy particularly for hydrometeor simulations with ice clouds in the far-IR.
- MFASIS-NN updates include improved accuracy by better treatment of water vapour and heterogenous surfaces, and code optimisation.

PC-RTTOV:

- New PC coefficients for IASI, IASI-NG, and Hamming apodised MTG-IRS supporting simulations over all surface types, with all trace gases, and optionally with either the NLTE correction, aerosol scattering, or hydrometeor scattering.
- Input profiles for PC-RTTOV simulations are no longer modified when the *apply_reg_limits* option is true. Values falling outside the limits are still flagged via the *radiance%quality(:)* output as for standard RTTOV simulations.

Wrapper:

- Python and C++ interfaces fully updated with respect to the changes in RTTOV.
- Enable return of explicit optical property Jacobians through wrapper.
- Enable user specification of radar K inputs so that the full Jacobian matrix can be computed for radar simulations.
- Add wrapper interface to the *rttov_aer_clim_prof* subroutine.
- Add a new *StoreEmisRefl* option and accessor functions to obtain surface emissivity/reflectance values used in the simulations so that the input emissivity/reflectance values are not overwritten.
- Rename *Options*, *Atlas*, and *Profiles* C++ source files with *Rttov* prefix to avoid potential name clashes/confusion with unrelated external libraries.
- Technical improvements to the C++ interface.

GUI:

- The RTTOV GUI is now a pure Python application that uses the *pyrttov* interface.
- The GUI now supports MW scattering simulations.
- The GUI no longer supports PC-RTTOV simulations.

Technical:

- **Numerous updates to the RTTOV Fortran interface to improve clarity and consistency. This includes significant changes to the *rttov_options* structure, updates to various module and subroutine names and subroutine interfaces, and other variable and derived type name changes.**
- HDF5 has been replaced by netCDF4 for large coefficient files and emissivity/BRDF atlas files. HDF5 is no longer an explicit dependence.
- New subroutine *rttov_wmo2rttov_sat_id* that maps WMO satellite IDs to RTTOV platform/satellite ID pairs.
- Reduce the number of memory allocations done within RTTOV to decrease run-time.
- Allow external allocation of all “trajectory” (internal state) data structures for single-threaded runs. This can improve performance in cases where many calls are made to RTTOV and the parallel interface is not used.

Capabilities removed:

- **Surface implicitly lies on bottom pressure half-level so RTTOV v14 cannot be called for profiles on fixed pressure levels with a separate surface pressure that is independent from the pressure levels.**
- FASTEM-1/2/3/4 and TESSEM2 microwave sea surface emissivity models.
- JONSWAP wave spectrum option for solar sea BRDF model.
- Solar single-scattering solver for clouds/aerosols.
- MFASIS-LUT fast visible solver for clouds based on look-up tables.
- HTFRTC Principal Components based model.
- Deprecated options removed: *grid_box_avg_cloud, dtau_test, reg_limit_extrap, spacetop*.

Coefficient file compatibility:

- RTTOV v14 can read all ASCII RTTOV v13-compatible optical depth coefficient files.
- Binary/Fortran unformatted optical depth coefficient files must be regenerated using RTTOV v14.
- HDF5 optical depth coefficient files cannot be read by v14: it is recommended to download the corresponding netCDF4 file from the RTTOV v14 coefficients download page.
- All other aerosol/hydrometeor optical property files and MFASIS-NN and PC-RTTOV coefficient files are mutually incompatible between v13 and v14: you must download new files from the RTTOV v14 coefficients download page.

3. Compilation and external dependencies

Full details of compilation are in section 5 of the RTTOV v14 user guide. Compilation of v14 is very similar to v13. The main difference is that netCDF replaces HDF5 as the sole (optional) external library dependency of the RTTOV Fortran code. Large coefficient files and the emissivity/BRDF atlas files for RTTOV v14 are distributed in netCDF format instead of HDF5. You must edit the *build/Makefile.local* with the location of your netCDF library. You may also need to specify the HDF5 library that was used to build your netCDF library.

The *src/mw_scatt/* source directory no longer exists, so no corresponding *librttov*_mw_scatt.a* is built during compilation. All replacement/equivalent user-level routines can be linked from *librttov14_main.a*, *librttov14_parallel.a*, *librttov14_coef_io.a*, and *librttov14_other.a*. See section 6 of the RTTOV v14 user guide.

4. Coefficient and optical property files

All RTTOV v13 ASCII gas optical depth coefficients are compatible with v14. Any binary (Fortran unformatted) optical depth coefficient files must be regenerated using the *rttov_conv_coef.exe* tool in RTTOV v14 (Annex A of the user guide). NetCDF replaces HDF5, so you must either convert ASCII files to netCDF using *rttov_conv_coef.exe* or download the netCDF equivalents of any HDF5 files from the RTTOV v14 coefficients download page.

In order to reduce the variety of coefficient files to a manageable number, only a subset of existing coefficient files are recommended for use with RTTOV v14. All other files are considered deprecated and will no longer be generated after the release of RTTOV v15, currently planned in early 2027. See section 3 of the user guide and the RTTOV coefficients download page for more information.

For SSU PMC shift optical depth coefficient files, you must now specify the cell pressure in *coefs%coef%pmc_coef%ppmc(1:nchan)*.

Aerosol and cloud/hydrometeor optical property files are mutually incompatible between RTTOV v13 and v14 and so RTTOV v14 compatible files must be downloaded from the RTTOV website. There is now a common format used for all optical properties and all sensors (UV/VIS/IR and MW). The new files have *rttov_aertable* and *rttov_hydrotable* prefixes. The optical properties available remain the same as in RTTOV v13. For hydrometeors, the MW hydrotable files are now handled in the same way as the UV/VIS/IR hydrotable files and so may be freely converted between ASCII, binary, and netCDF formats, and data for subsets of channels may be extracted, just as for other RTTOV coefficient files. The hydrotable files for all sensors are read via the same arguments to the *rttov_read_coefs* subroutine.

The MW ARO scaling polarisation look up table file (*ScalingFactorForBulkProperties.rssp*) remains the same as in v13. This can be read at the same time as other coefficients via *rttov_read_coefs* subroutine.

MFASIS-NN coefficients files are mutually incompatible between RTTOV v13 and v14 and so RTTOV v14 files must be downloaded from the website. In v14 the MFASIS-NN filenames have the prefix *rttov_mfasis_nn_hydro_* (note the additional "*hydro_*") to distinguish them from future aerosol MFASIS-NN coefficient files.

PC-RTTOV coefficient files are mutually incompatible between RTTOV v13 and v14 and so RTTOV v14 files must be downloaded from the website along with the associated optical depth coefficient files.

As noted above, large files are now distributed in netCDF format instead of HDF5. HDF5 is no longer a supported file format in RTTOV.

5. Use of named constants

There are various occasions where the RTTOV interface requires a selection among multiple integer options. This occurs in several options (section 8) and profile variables (section 9), for example. It is recommended to use the integer constants defined in the *rttov_const* module instead of using integer literals in your code where possible. This will make your code more readable, and it also ensures that it continues to work if the underlying integer values change in subsequent releases.

The same applies in other circumstances, such as the indices for the various aerosol and hydrometeor particle types for which pre-defined optical properties are provided.

Some of these constants are listed in this document. Annex K of the RTTOV v14 user guide provides a full list of all constants relevant to user code. Some of these constants existed in RTTOV v13 but were not documented, and others have been added or renamed in v14.

6. Subroutine interfaces and executables

There are changes to many user-level subroutines. Some have changed name, and the interfaces to many have changed in various ways. Full details of all user-level subroutine interfaces and RTTOV executables are given in the Annexes in the RTTOV v14 user guide. This section summarises the changes to subroutines and executables. For reference, the interfaces of the corresponding routines in RTTOV v13 are given in the Annexes of the RTTOV v13 user guide.

When updating your code, it is recommended to check the lists below to identify which of the RTTOV subroutines you are calling have changed, and then to look at the relevant Annex in the user guide to see the updated subroutine interface.

For all subroutines that take an *nlevels* argument, this refers to the number of pressure half-levels (see section 9). For most applications this is one larger than *nlevels* in v13.

Wherever the coefficients structure is an argument (dummy argument *coefs*), it is the top-level *rttov_coefs* structure that is required which contains all optical depth, MFASIS-NN, and PC-RTTOV coefficients, and aerosol/hydrometeor optical property data read in via a call to the *rttov_read_coefs* subroutine.

Section 7 gives further general information about the changes to the subroutines used to allocate/deallocate RTTOV data structures.

Subroutines with changed names

- *rttov_alloc_prof* – renamed *rttov_alloc_profiles*
- *rttov_init_prof* – renamed *rttov_init_profiles*
- *rttov_alloc_rad* – renamed *rttov_alloc_radiance*
- *rttov_init_rad* – renamed *rttov_init_radiance*
- *rttov_calc_legcoef* – renamed *rttov_calc_lcoef*
- *rttov_user_options_checkinput* – renamed *rttov_user_check_options*
- *rttov_user_profile_checkinput* – renamed *rttov_user_check_profile*
- *rttov_scatt_emis_retrieval* – renamed *rttov_emissivity_retrieval*

Subroutines with changed interfaces

See the Annexes of the RTTOV v14 user guide for full details of all subroutine interfaces.

- *rttov_direct, rttov_tl, rttov_ad, rttov_k* (Annex H)
- *rttov_parallel_direct, rttov_parallel_tl, rttov_parallel_ad, rttov_parallel_k* (Annex H: same interfaces as *rttov_direct/tl/ad/k* but with additional optional *nthreads* argument)
- *rttov_read_coefs* (Annex C)
- *rttov_alloc_direct, rttov_alloc_tl, rttov_alloc_ad, rttov_alloc_k* (Annex D)
- *rttov_alloc_profiles, rttov_alloc_transmission, rttov_alloc_radiance* (Annex D)
- *rttov_alloc_opt_param, rttov_alloc_reflectivity, rttov_alloc_emis_ret_terms* (Annex D)
- *rttov_alloc_pccomp* (Annex D)
- *rttov_alloc_traj* (this still exists but you should call *rttov_alloc_traj_all* instead, Annex D)
- *rttov_init_emis_refl, rttov_init_opt_param* (Annex D)
- *rttov_bpr_calc* (Annex E)
- *rttov_setup_emis_atlas, rttov_get_emis, rttov_get_sea_emis* (Annex F)
- *rttov_get_brdf, rttov_get_sea_brdf* (Annex G)
- *rttov_user_check_options, rttov_user_check_profile* (Annex I)
- *rttov_calc_geo_sat_angles* (Annex I)
- *rttov_aer_clim_prof* (Annex I)
- *rttov_emissivity_retrieval* (Annex I)
- *rttov_get_pc_predictindex* (Annex I)

Deleted subroutines and v14 equivalents

- *rttov_scatt* - run MW scattering direct simulations using *rttov_direct*
- *rttov_scatt_tl* - run MW scattering TL simulations using *rttov_tl*
- *rttov_scatt_ad* - run MW scattering AD/K simulations using *rttov_ad/k*
- *rttov_parallel_scatt* - run multi-threaded MW scattering direct simulations using *rttov_parallel_direct*
- *rttov_parallel_scatt_tl* - run multi-threaded MW scattering TL simulations using *rttov_parallel_tl*
- *rttov_parallel_scatt_ad* - run multi-threaded MW scattering AD/K simulations using *rttov_parallel_ad/k*
- *rttov_read_scattcoeffs* - read hydrotable files using *rttov_read_coefs*
- *rttov_dealloc_scattcoeffs* - deallocate hydrotable data using *rttov_dealloc_coefs*
- *rttov_alloc_scatt_prof* - allocate profile structure using *rttov_alloc_profiles*

- *rttov_init_scatt_prof* - initialise profile structure using *rttov_init_profiles*
- *rttov_print_opts_scatt* - print options structure using *rttov_print_opts*
- *rttov_print_cld_profile* - print profile structure using *rttov_print_profile*
- *rttov_scatt_setupindex* - no longer required (mapping of channels to optical property tables is handled internally by RTTOV).
- *rttov_read_coefs_htfrtc* - HTFRTC no longer supported.

New subroutines

- *rttov_alloc_emis_refl* – for allocating the new *rttov_emis_refl* structure (see section 11). Interface in Annex D of user guide.
- *rttov_alloc_traj_all* – this should be used instead of *rttov_alloc_traj* (see section 7). Interface in Annex D.
- *rttov_init_opt_param_solar* – this must be called before running RTTOV for scattering simulations with explicit optical properties when solar radiation is enabled to pre-compute some phase function related data (see section 12). Previously this was done using the *rttov_init_opt_param* subroutine. Interface in Annex D.
- *rttov_init_emis_ret_terms* – initialises the members of an *rttov_emis_retrieval_terms* structure (see section 12). Interface in Annex D.
- *rttov_alloc_diag_output* – allocates the new *rttov_diagnostic_output* structure (see section 14). Interface in Annex D.
- *rttov_init_diag_output* – initialises the members of an *rttov_diagnostic_output* structure. Interface in Annex D.
- *rttov_calc_asym* – calculates asymmetry parameter given a phase function (see section 12). Interface in Annex E.
- *rttov_user_check_emis_refl* – checks an *rttov_emis_refl* structure for valid inputs (see section 10). Interface in Annex I.
- *rttov_user_check_opt_param* – checks an explicit optical property structure for valid inputs (see section 10). Interface in Annex I.
- *rttov_wmo2rttov_sat_id* – maps WMO satellite IDs to RTTOV platform and satellite ID couplets. Interface in Annex I.

Executables with changed names and/or arguments

- *rttov_conv_coef.exe* – this is used to convert gas absorption optical depth, MFASIS-NN, and PC-RTTOV coefficient files, and aerosol/hydrometeor optical property files between ASCII, Fortran unformatted (“binary”), and netCDF formats. It also allows data for channel subsets to be extracted from any of these file types (Annex A of RTTOV v14 user guide).
- *rttov_make_scaercoef.exe* – renamed *rttov_make_aertable.exe* (Annex I).
- *rttov_obs_to_pc.exe* – some arguments have been renamed (Annex I).

Deleted executables

- *rttov_scatt_ascii2bin_scattcoef.exe* – use *rttov_conv_coef.exe* instead.
- *rttov_mafsis_lut_info.exe* – redundant as MFASIS-LUT is no longer implemented.
- *rttov11_conv_coef_11to12.exe* – redundant as RTTOV v11/v12-era coefficients files are deprecated.

- *rttov11_conv_coef_12to11.exe* – redundant as RTTOV v11/v12-era coefficients files are deprecated.

7. Kinds, derived types, and allocation/deallocation subroutines

The module containing the KINDs used by RTTOV was previously called *parkind1* and has been renamed *rttov_kinds*. The integer and logical KINDs required in the user interface remain unchanged: these are *jpim* and *jplm* respectively. The real KIND has been renamed from *jprb* to *jprv*.

All user-level derived types are described fully in Annex J of the RTTOV v14 user guide, and the (de)allocation subroutines are described in Annex D. These may be compared to the RTTOV v13 equivalents in Annex O of the RTTOV v13 user guide for the derived types, and Annex D for the (de)allocation subroutines.

Derived types with changed names

- *rttov_scatt_emis_retrieval_type* – renamed *rttov_emis_retrieval_terms.*

The emissivity and BRDF atlas derived types (*rttov_emis_atlas_data, rttov_brdf_atlas_data*) have not changed, but the modules in which they are contained have been renamed:

- *mod_rttov_emis_atlas* renamed *rttov_emis_atlas_mod*
- *mod_rttov_brdf_atlas* renamed *rttov_brdf _atlas_mod*

Derived types with changes to member variables

- *rttov_options* – substantial changes, see section 8.
- *rttov_profile* – see section 9.
- *rttov_opt_param* – *abs* and *sca* members replaced by *ext* (total extinction) and *ssa* (single-scattering albedo), addition of *asym* member (asymmetry parameter for the delta-Eddington thermal solver), and *legcoef* member renamed *lcoef.* See section 12.
- *rttov_radiance* – *plane_parallel* member removed and *geometric_height* member now available in new *rttov_diagnostic_output* structure. See section 14.
- *rttov_emis_retrieval_terms* – significant changes, see Annex J in RTTOV v14 user guide.
- *rttov_pccomp* – members that were only populated by the HTFRTC model have been removed. See section 13.

Deleted derived types

- *rttov_emissivity, rttov_reflectance* – merged into *rttov_emis_refl*
- *rttov_options_scatt* – RTTOV-SCATT options have been merged into *rttov_options*
- *rttov_profile_cloud* – RTTOV-SCATT profile data have been merged into *rttov_profile*
- *rttov_scatt_coeff* – the MW hydrometeor optical properties are stored within *rttov_coefs*

New derived types

- *rttov_emis_refl* – new structure containing input/output surface emissivity and reflectance related data. See section 11.

- *rttov_diagnostic_output* – new structure containing per-profile outputs including geometric heights of pressure levels and effective hydrometeor fraction for hydrometeor scattering simulations. See section 14.

Changes to allocation/deallocation subroutines

The interfaces to all user-level (de)allocation subroutines (*rttov_alloc_**) have been changed to improve clarity and consistency (see Annex D of RTTOV v14 user guide).

The previous integer "*asw"* argument to these subroutines in RTTOV v13 that was used to select between allocation and deallocation has been replaced by a logical *alloc* argument which should be true to allocate and false to deallocate. The *alloc_flag* and *dealloc_flag* constants in *rttov_const* are recommended over logical literals. The *alloc* argument is always the second argument following the *err* return status argument.

Aside from the *rttov_alloc_traj_all* subroutine (see below), the structure being (de)allocated is always the third argument to the subroutines for (de)allocating individual data structures (i.e., excluding the *rttov_alloc_direct/tl/ad/k* subroutines that are used for allocating multiple data structures together).

The integer dimension arguments (specifying array sizes) are always in a consistent order between the subroutines which is different in many cases to the order in RTTOV v13. The dimension sizes are typically the same as for RTTOV v13 except for *nlevels* which specifies the number of pressure half-levels in RTTOV v14 (see section 9) which is one larger than *nlevels* in RTTOV v13 for most applications.

The optional logical *init* argument to most of the allocation routines can be set to true to initialise the members of the newly allocated structure. This is always the final argument where it exists.

The argument order for the *rttov_alloc_direct/tl/ad/k* subroutines follows the argument order for the corresponding top-level RTTOV routines *rttov_direct/tl/ad/k*.

Some allocation routines have an *opts* argument. In this case, some members of the structure will only be allocated if certain options are set (such as *opts%rt_all%solar* to enable solar radiation, or *opts%scatt%aerosols* or *opts%scatt%hydrometeors* to enable scattering simulations). It is important to set the options you require before calling the allocation subroutines.

Some allocation routines have a *coefs* argument. This must be populated by a call to *rttov_read_coefs* before calling the allocation subroutine.

Some allocation routines have an optional logical *direct* argument. The TL/AD/K counterparts to some direct model arguments only require a subset of members to be allocated (one example is the *radiance%quality(:)* member which is only populated by the direct model). When allocating the TL/AD/K variables (e.g., *radiance_tl/ad/k*) you can set the *direct* argument to false to avoid allocating unnecessary arrays, thereby minimising memory requirements. For the corresponding direct model variables (e.g., *radiance*) you can either set *direct* to true or omit the argument to allocate all relevant members of the structure. The *direct* argument may be omitted in all cases without causing any problems.

The *rttov_alloc_traj_all* subroutine should now be used instead of the *rttov_alloc_traj* subroutine. This allows all three "trajectory" structures to be allocated outside of RTTOV instead of them being allocated inside RTTOV which can be slower when making many calls to RTTOV. Some limitations apply to the use of these trajectory arguments to RTTOV as described in section 7.3.2 of the user guide, but in cases where the RTTOV parallel interface (user guide section 7.10) is not being used this is generally recommended for most types of simulation.

8. Options

The *rttov_options* structure is used to configure simulations at run-time. Options are grouped into sub-types according to function. These structures have been significantly reorganised, and many options renamed in v14.0 to eliminate (as much as possible) spectral distinctions, to eliminate naming inconsistencies, and to improve clarity. Table 8.1 compares the options sub-types in v13.2 with those in v14.0. Table 8.2 lists the v13.2 options and gives the v14.0 equivalents. Table 8.3 describes all v14.0 options and gives the v13.2 equivalents. **Note that some option default values have changed in v14.0.**

For integer options that select among parameterisations/models, Table 8.4 gives the integer constants that are recommended for use when setting these options in user code (see section 5 above, and all integer constants can be found in Annex K of the RTTOV v14 user guide). Some v13.x options have been removed as they are no longer relevant in v14.0 or were previously deprecated. There is no longer a separate *rttov_options_scatt* structure for MW scattering simulations.

Table 8.1: option sub-types in v13.x and v14.0

RTTOV v13.x sub-types	**Description**
opts % config	General configuration options
opts % interpolation	Interpolation options
opts % rt_all	General RT options
opts % rt_mw	MW RT options
opts % rt_ir	UV/VIS/IR RT options
opts % rt_ir % pc	PC-RTTOV options
opts % htfrtc_opts	HTFRTC options
opts_scatt	Separate structure for RTTOV-SCATT options
RTTOV v14.0 sub-types	**Description**
opts % config	General configuration options
opts % interpolation	Interpolation options
opts % rt_all	General RT options
opts % surface	Surface-related options
opts % clw_absorption	Options for non-scattering cloud liquid water absorption
opts % scatt	Scattering options
opts % cloud_overlap	Options for cloud overlap
opts % pcrttov	PC-RTTOV options

Table 8.2: RTTOV v13.x options and their v14.0 equivalents. Those in red have changed since v13.x, by moving to a different sub-structure and/or being renamed or being removed.

RTTOV v13.2 option	RTTOV v14.0 equivalent
opts % config % verbose	opts % config % verbose
opts % config % do_checkinput	opts % config % check_profiles
opts % config % apply_reg_limits	opts % config % apply_reg_limits
opts % config % opdep13_gas_clip	opts % config % opdep13_gas_clip
opts % config % fix_hgpl	N/A
opts % interpolation % addinterp	opts % interpolation % enable_interp
opts % interpolation % interp_mode	opts % interpolation % interp_mode
opts % interpolation % lgradp	opts % interpolation % pressure_gradients
opts % interpolation % spacetop	N/A
opts % interpolation % reg_limit_extrap	N/A
opts % rt_all % ozone_data	opts % rt_all % o3_data
opts % rt_all % co2_data	opts % rt_all % co2_data
opts % rt_all % n2o_data	opts % rt_all % n2o_data
opts % rt_all % co_data	opts % rt_all % co_data
opts % rt_all % ch4_data	opts % rt_all % ch4_data
opts % rt_all % so2_data	opts % rt_all % so2_data
opts % rt_all % addrefrac	opts % rt_all % refraction
opts % rt_all % plane_parallel	opts % rt_all % plane_parallel
opts % rt_all % rad_down_lin_tau	opts % rt_all % rad_down_lin_tau
opts % rt_all % use_t2m_opdep	opts % rt_all % use_t2m
opts % rt_all % use_q2m	opts % rt_all % use_q2m
opts % rt_all % use_tskin_eff	opts % surface % use_tskin_eff
opts % rt_all % do_lambertian	opts % surface % lambertian
opts % rt_all % lambertian_fixed_angle	opts % surface % lambertian_fixed_angle
opts % rt_all % switchrad	opts % config % adk_bt
opts % rt_all % transmittances_only	opts % config % transmittances_only
opts % rt_all % dtau_test	N/A
opts % rt_mw % clw_data	opts % clw_absorption % clw_data
opts % rt_mw % clw_scheme	opts % clw_absorption % permittivity_param
opts % rt_mw % clw_cloud_top	opts % clw_absorption % clw_cloud_top
opts % rt_mw % fastem_version	opts % surface % mw_sea_emis_model
opts % rt_mw % fastem3_rwd_fix	N/A
opts % rt_mw % supply_foam_fraction	opts % surface % use_foam_fraction
opts % rt_ir % addsolar	opts % rt_all % solar
opts % rt_ir % rayleigh_max_wavelength	opts % rt_all % rayleigh_max_wavelength
opts % rt_ir % rayleigh_min_pressure	opts % rt_all % rayleigh_min_pressure
opts % rt_ir % rayleigh_single_scatt	opts % rt_all % rayleigh_single_scatt
opts % rt_ir % rayleigh_depol	N/A
opts % rt_ir % do_nlte_correction	opts % rt_all % nlte_correction
opts % rt_ir % solar_sea_brdf_model	opts % surface % solar_sea_refl_model
opts % rt_ir % ir_sea_emis_model	opts % surface % ir_sea_emis_model
opts % rt_ir % addaerosl	opts % scatt % aerosols
opts % rt_ir % addclouds	opts % scatt % hydrometeors
opts % rt_ir % user_aer_opt_param	opts % scatt % user_aer_opt_param
opts % rt_ir % user_cld_opt_param	opts % scatt % user_hydro_opt_param
opts % rt_ir % ir_scatt_model	opts % scatt % thermal_solver
opts % rt_ir % vis_scatt_model	opts % scatt % solar_solver
opts % rt_ir % dom_rayleigh	opts % scatt % rayleigh_multi_scatt
opts % rt_ir % dom_nstreams	opts % scatt % dom_nstreams

opts % rt_ir % dom_accuracy	opts % scatt % dom_accuracy
opts % rt_ir % dom_opdep_threshold	opts % scatt % dom_opdep_threshold
opts % rt_ir % cloud_overlap	opts % cloud_overlap % overlap_param
opts % rt_ir % grid_box_avg_cloud	N/A
opts % rt_ir % cldcol_threshold	opts % cloud_overlap % col_threshold
opts % rt_ir % cc_low_cloud_top	opts % cloud_overlap % two_col_max_frac_max_p
opts % rt_ir % pc % addpc	opts % pcrttov % enable_pcrttov
opts % rt_ir % pc % npcscores	opts % pcrttov % npcscores
opts % rt_ir % pc % addradrec	opts % pcrttov % rec_rad
opts % rt_ir % pc % ipcbnd	opts % pcrttov % pc_band
opts % rt_ir % pc % ipcreg	opts % pcrttov % pc_reg_set
opts % dev % do_opdep_calc	opts % config % gas_opdep_calc
opts_scatt % config	opts % config
opts_scatt % interp_mode	opts % interpolation % interp_mode
opts_scatt % lgradp	opts % interpolation % pressure_gradients
opts_scatt % reg_limit_extrap	N/A
opts_scatt % ozone_data	opts % rt_all % o3_data
opts_scatt % addrefrac	opts % rt_all % refraction
opts_scatt % rad_down_lin_tau	opts % rt_all % rad_down_lin_tau
opts_scatt % use_t2m_opdep	opts % rt_all % use_t2m
opts_scatt % use_q2m	opts % rt_all % use_q2m
opts_scatt % dtau_test	N/A
opts_scatt % fastem_version	opts % surface % mw_sea_emis_model
opts_scatt % fastem3_rwd_fix	N/A
opts_scatt % supply_foam_fraction	opts % surface % use_foam_fraction
opts_scatt % lusercfrac	Selected via opts % cloud_overlap % overlap_param
opts_scatt % cc_threshold	opts % cloud_overlap % col_threshold
opts_scatt % pol_mode	opts % scatt % mw_pol_mode
opts_scatt % ice_polarisation	opts % scatt % ice_polarisation
opts_scatt % hydro_cfrac_tlad	opts % cloud_overlap % hydro_frac_tlad
opts_scatt % zero_hydro_tlad	opts % scatt % zero_hydro_tlad

Table 8.3: RTTOV v14.0 options and their v13.x equivalents. See Table 8.4 for valid integer option values. Options shaded in grey should be left on their default values unless you have a specific reason to change them.

Option	**Type**	**Brief description**	**v13.x equivalent**
opts % config %			
verbose	Logical	If false only messages for fatal errors are output (default = false).	opts % config % verbose
apply_reg_limits	Logical	If true, profile values outside the limits of the training data for the gas optical depth regression are clipped to the limits before input to the regression (default = false).	opts % config % apply_reg_limits
check_profiles	Logical	If true, input profiles are checked for unphysical values and whether they lie within the regression limits (default = true). NB in v14 profiles are always compared to the min/max regression limits regardless of this option, unlike in v13 where if *do_checkinput* is false this is not done.	opts % config % do_checkinput
transmittances_only	Logical	Direct model only: if true only transmittances are calculated. Output radiances, and surface emissivities and reflectances are zero. This is	opts % rt_all % transmittances_only

		more efficient if only transmittances are required (default = false).	
bt_overcast_calc	Logical	If true, compute overcast brightness temperatures in *radiance%bt_overcast* output array (default = False).	N/A
gas_opdep_calc	Logical	If false, disables the gas optical depth calculation and simulations are run assuming zero optical depths. Intended for users investigating certain scattering scenarios (default = true).	opts % dev % do_opdep_calc
adk_bt	Logical	Determines input increment for AD/K routines: if true *radiance_ad/k % bt* is used for thermal channels, otherwise *radiance_ad/k % total* is used (default = true)	opts % rt_all % switchrad
adk_refl	Logical	Determines input increment for AD/K routines: if true *radiance_ad/k % refl* is used for non-thermal channels, otherwise *radiance_ad/k % total* is used (default = true)	N/A
opdep13_gas_clip	Logical	If true, negative layer optical depths for individual gases are reset to zero in the v13 predictor calculations. This can cause convergence problems in some DA systems in which case this can be set to false (default = true).	opts % config % opdep13_gas_clip
opts % interpolation %			
enable_interp	Logical	If true, input profiles may be input on arbitrary pressure levels and they are interpolated by RTTOV for the gas optical depth regression. Should be true in most use cases in v14 (default = true).	opts % interpolation % addinterp
interp_mode	Integer	Select interpolation mode, only applies if *enable_interp* is true (default = *interp_rochon_wfn*).	opts % interpolation % interp_mode
pressure_gradients	Logical	If true, the profile *p_half* and *p* members are active variables in the TL/AD/K, only applies if *enable_interp* is true (default = false).	opts % interpolation % lgradp
opts % rt_all %			
o3_data	Logical	Set to true if supplying O_3 profiles to RTTOV (default = false). Coefficient file must support variable O_3.	opts % rt_all % ozone_data
co2_data	Logical	Set to true if supplying CO_2 profiles to RTTOV (default = false). Coefficient file must support variable CO_2.	opts % rt_all % co2_data
n2o_data	Logical	Set to true if supplying N_2O profiles to RTTOV (default = false). Coefficient file must support variable N_2O.	opts % rt_all % n2o_data
co_data	Logical	Set to true if supplying CO profiles to RTTOV (default = false). Coefficient file must support variable CO.	opts % rt_all % co_data
ch4_data	Logical	Set to true if supplying CH_4 profiles to RTTOV (default = false). Coefficient file must support variable CH_4.	opts % rt_all % ch4_data
so2_data	Logical	Set to true if supplying SO_2 profiles to RTTOV (default = false). Coefficient file must support variable SO_2.	opts % rt_all % so2_data
solar	Logical	Enable solar radiation in simulations (default = false).	opts % rt_ir % addsolar
rayleigh_max_wavelength	Real	Specify maximum channel wavelength for which to enable Rayleigh scattering (units = microns, default = 2 microns). Set to zero to disable	opts % rt_ir % rayleigh_max_wavelength

		Rayleigh extinction and scattering entirely with v13 predictor coefficients.	
rayleigh_min_pressure	Real	Rayleigh scattering is ignored in layers at pressures below this (units = hPa, default = 0 hPa).	opts % rt_ir % rayleigh_min_pressure
rayleigh_single_scatt	Logical	Enable Rayleigh single scattering parameterisation. If false, Rayleigh extinction is still included in the simulations. The *rayleigh_multi_scatt* option overrides this for DOM simulations (default = true).	opts % rt_ir % rayleigh_single_scatt
nlte_correction	Logical	Enable the NLTE bias correction (default = false). Coefficient file must support the NLTE correction.	opts % rt_ir % do_nlte_correction
refraction	Logical	Enable atmospheric refraction in simulations (default = true). This is ignored for solvers that assume strict plane parallel geometry or if the *plane_parallel* option is true.	opts % rt_all % addrefrac
plane_parallel	Logical	Enable strict plane parallel geometry (no Earth curvature). It is not necessary to set this for solvers that assume plane parallel geometry (default = false).	opts % rt_all % plane_parallel
rad_down_lin_tau	Logical	Use linear-in-tau (true) or layer-average (false) for downwelling layer emission in radiative transfer equation integration (default = true). The latter is slightly faster and has negligible impact on radiances. Applies to clear-sky calculations, and to the Chou-scaling thermal scattering solver. Lambertian downwelling radiances always use the layer-average.	opts % rt_all % rad_down_lin_tau
use_t2m	Logical	Enable use of the 2m temperature profile variable (default = true).	opts % rt_all % use_t2m_opdep
use_q2m	Logical	Enable use of the 2m water vapour profile variable (default = true).	opts % rt_all % use_q2m
opts % surface %			
solar_sea_refl_model	Integer	Select solar reflectance model for ocean surfaces (default = *solar_refl_model_elfouhaily*). There is currently only one option.	opts % rt_ir % solar_sea_brdf_model
ir_sea_emis_model	Integer	Select IR emissivity model for ocean surfaces (default = *ir_emis_model_iremis*).	opts % rt_ir % ir_sea_emis_model
mw_sea_emis_model	Integer	Select MW emissivity model for ocean surfaces (default = *mw_emis_model_surfem_ocean*).	opts % rt_mw % fastem_version
use_foam_fraction	Logical	Enable use of user-supplied foam fraction in emissivity models, where supported (default = false).	opts % rt_mw % supply_foam_fraction
lambertian	Logical	Enable Lambertian surface option which allows surfaces to be treated as a configurable mixture of Lambertian and specular reflectors (default = false).	opts % rt_all % do_lambertian
lambertian_fixed_angle	Logical	Use fixed effective zenith angle (true) or parameterised angle (false) when computing Lambertian downwelling radiances (default = true).	opts % rt_all % lambertian_fixed_angle
use_tskin_eff	Logical	If true, use per-channel effective skin temperatures input via *emis_refl(:)%tskin_eff(:)* instead of *profiles(:)%skin(:)%t*	opts % rt_all % use_tskin_eff
opts % clw_absorption %			
clw_data	Logical	Activate cloud liquid water absorption. This is not compatible with the *hydrometeor* option (default = false).	opts % rt_mw % clw_data

permittivity_param	Integer	Select liquid water permittivity parameterisation (default = clw_perm_rosenkranz).	opts % rt_mw % clw_scheme
clw_cloud_top	Real	Lower pressure limit for CLW absorption calculations, i.e., CLW is ignored in layers at pressures lower than this (units = hPa, default = 322 hPa).	opts % rt_mw % clw_cloud_top
opts % scatt %			
hydrometeors	Logical	Enable scattering by hydrometeors (default = false).	opts % rt_ir % addclouds
aerosols	Logical	Enable scattering by aerosols (default = false).	opts % rt_ir % addaerosl
thermal_solver	Integer	Select solver for thermal radiation (default = *thermal_solver_delta_edd*).	opts % rt_ir % ir_scatt_model
solar_solver	Integer	Select solver for solar radiation (default = *solar_solver_dom*).	opts % rt_ir % vis_scatt_model
radar	Logical	Enable radar reflectivity simulations alongside passive radiance simulations (default = false).	N/A
user_hydro_opt_param	Logical	Enable explicit input of hydrometeor optical property profiles per channel (default = false).	opts % rt_ir % user_cld_opt_param
user_aer _opt_param	Logical	Enable explicit input of aerosol optical property profiles per channel (default = false).	opts % rt_ir % user_aer_opt_param
baran_ice_version	Integer	Select version of Baran ice scheme (default = *baran2018*).	N/A
rayleigh_multi_scatt	Logical	Enable Rayleigh multiple scattering (default = false). Currently only applicable with *solar_solver=solar_solver_dom*.	opts % rt_ir % dom_rayleigh
dom_nstreams	Integer	Select number of streams for simulations using the DOM solver (default = 8).	opts % rt_ir % dom_nstreams
dom_accuracy	Real	Convergence criterion for termination of DOM azimuthal loop for solar DOM simulations (default = 0 i.e. no early termination).	opts % rt_ir % dom_accuracy
dom_opdep_threshold	Real	DOM simulations ignore layers at level-to-space absorption optical depths larger than this, not applied if <= 0 (default = 0).	opts % rt_ir % dom_opdep_threshold
chou_tang_mod	Logical	Enable Tang modification to Chou-scaling (default = false).	N/A
chou_tang_factor	Real	Empirical factor used in Tang modification to Chou-scaling (default = 0.1).	N/A
mw_pol_mode	Integer	Select treatment of polarised scattering (default = *mw_pol_mode_empirical*). Only applies to MW sensors.	opts_scatt % pol_mode
ice_polarisation	Real	Polarised scattering factor for ice hydrometeors for empirical polarisation *mw_pol_mode* (default = 1.4). Only applies to MW sensors.	opts_scatt % ice_polarisation
zero_hydro_tlad	Logical	Enable hydrometeor TL/AD/K sensitivity in layers with zero hydrometeor concentration (default = false). Currently only applies to MW sensors and two-column overlap schemes with the delta-Eddington thermal solver.	opts_scatt % zero_hydro_tlad
opts % cloud_overlap % *- only relevant when opts % scatt % hydrometeors is true*			
overlap_param	Integer	Select cloud overlap parameterisation (default = *cloud_overlap_auto_select*).	opts % rt_ir % cloud_overlap
per_hydro_frac	Logical	Enable input of separate *hydro_frac* for each hydrometeor. When false, a single *hydro_frac* applies per layer with all hydrometeors well mixed in that fraction (default = false). Can only be true for radar simulations, and for the *cloud_overlap_2col_weighted* overlap scheme.	N/A

col_threshold	Real	Cloud columns with weights (multi-column overlap) or effective cloud fractions (2-column overlap) smaller than this value are treated as clear (default = 0 i.e. retain all columns). Value must be in range 0-1.	opts % rt_ir % cldcol_threshold AND opts_scatt % cc_threshold
two_col_max_frac_max_p	Real	For *cloud_overlap_2col_max_frac*, this is the maximum pressure applied when determining the effective cloud fraction (units = hPa, default = 750 hPa).	opts % rt_ir % cc_low_cloud_top
hydro_frac_tlad	Logical	Switch to enable/disable hydrometeor TL/AD/K sensitivity to *hydro_frac*. NB setting false breaks the consistency between the direct and TL/AD/K models (default = true). If true, ignored unless the *cloud_overlap_2col_weighted* overlap scheme is used.	opts_scatt % hydro_cfrac_tlad
opts % pcrttov %			
enable_pcrttov	Logical	Enable PC-RTTOV simulations (default = false).	opts % rt_ir % pc % addpc
npcscores	Integer	Number of PC scores to simulate (default = -1). The maximum number depends on the instrument being simulated. Minimum value is 1.	opts % rt_ir % pc % npcscores
rec_rad	Logical	If false output only PC scores, if true compute reconstructed radiances and BTs (default = false).	opts % rt_ir % pc % addradrec
pc_band	Integer	Select PC spectral band (default = 1). Valid values depend on the PC coefficient file. For current coefficients this should always be 1.	opts % rt_ir % pc % ipcbnd
pc_reg_set	Integer	Select PC regression channel set (default = -1). Valid values depend on the PC coefficient file and vary between instruments. Minimum value is 1.	opts % rt_ir % pc % ipcreg

Some options are integer values and allow selection among different parameterisations, models, or calculation methods. In RTTOV v13.x the integer values were sometimes explicitly used to select among the available choices. In RTTOV v14.0 integer constants are defined in the *rttov_const* module for all available options (Annex K of the user guide), and it is recommended to use these in preference to the explicit integer values. This makes code self-documenting and ensures that it will continue to work if the underlying integer values change in future RTTOV releases. Table 8.4 lists the integer constants available for each integer option.

Table 8.4: named integer constants defined in rttov_const recommended to be used when setting integer options, and their v13.x equivalents.

Option	**Integer constants defined in *rttov_const***	**Description**
opts % interpolation % **interp_mode**	interp_rochon (1)	Rochon interpolation of profiles and optical depths (in v13.x *interp_mode* = 1)
	interp_loglinear (2)	Log-linear interpolation of profiles and optical depths (in v13.x *interp_mode* = 2)
	interp_rochon_loglinear (3)	Rochon interpolation of profiles and log-linear interpolation of optical depths (in v13.x *interp_mode* = 3)
	interp_rochon_wfn (4)	Rochon interpolation of profiles and weighting functions (in v13.x *interp_mode* = 4)

	interp_rochon_loglinear_wfn (5)	Rochon interpolation of profiles and log-linear interpolation of weighting functions (in v13.x *interp_mode* = 5)
opts % surface % **solar_sea_refl_model**	solar_refl_model_elfouhaily (1)	Elfouhaily wave spectrum parameterisation for solar sea surface BRDF model (in v13.x *solar_sea_brdf_model* = 2)
opts % surface % **ir_sea_emis_model**	ir_emis_model_isem (1)	ISEM IR sea surface emissivity model (in v13.x *ir_sea_emis_model* = 1)
	ir_emis_model_iremis (2)	IREMIS IR sea surface emissivity model (in v13.x *ir_sea_emis_model* = 2)
opts % surface % **mw_sea_emis_model**	mw_emis_model_fastem5 (1)	FASTEM-5 MW sea surface emissivity model (in v13.x *fastem_version* = 5)
	mw_emis_model_fastem6 (2)	FASTEM-6 MW sea surface emissivity model (in v13.x *fastem_version* = 6)
	mw_emis_model_surfem_ocean (3)	SURFEM-Ocean MW sea surface emissivity model (in v13.2 *fastem_version* = 7)
opts % clw_absorption % **permittivity_param**	clw_perm_liebe (1)	Liebe (1989) permittivity parameterisation (in v13.x *clw_scheme* = 1)
	clw_perm_rosenkranz (2)	Rosenkranz (2015) permittivity parameterisation (in v13.x *clw_scheme* = 2)
	clw_perm_tkc (3)	Turner, Kneifel, Cadeddu (2016) permittivity parameterisation (in v13.x *clw_scheme* = 3)
opts % scatt % **thermal_solver**	thermal_solver_dom (1)	DOM thermal solver (in v13.x *ir_scatt_model* = 1)
	thermal_solver_chou (2)	Chou-scaling solver (in v13.x *ir_scatt_model* = 2)
	thermal_solver_delta_edd (3)	Delta-Eddington solver (the v13.x RTTOV-SCATT solver)
opts % scatt % **solar_solver**	solar_solver_dom (1)	DOM solar solver (in v13.x *vis_scatt_model* = 1)
	solar_solver_mfasis_nn (2)	MFASIS neural-network solver (in v13.2 *vis_scatt_model* = 4)
opts % scatt % **baran_ice_version**	baran2014 (1)	Baran 2014 ice cloud scheme (in v13.x profile *ice_scheme* = 2)
	baran2018 (2)	Baran 2018 ice cloud scheme, recommended (in v13.x profile *ice_scheme* = 3)
opts % scatt % **mw_pol_mode**	mw_pol_mode_no_pol (0)	Disable polarised scattering (in v13.2 *pol_mode* = 0)
	mw_pol_mode_empirical (1)	Empirical scaling of extinction for V and H polarised channels, scale factor controlled by *ice_polarisation* option (in v13.2 *pol_mode* = 1)
	mw_pol_mode_aro_scaling (2)	ARO scaled polarisation for V, H, QV, QH polarised channels, requires additional polarisation coef file when reading coefficients (in v13.2 *pol_mode* = 2)
opts % cloud_overlap % **overlap_param**	cloud_overlap_auto_select (0)	Automatic selection of *overlap_param* at run-time: for VIS/IR sensors use max/random and for MW sensors use two-column hydro-weighted scheme (same RTTOV and RTTOV-SCATT defaults as in v13.x)
	cloud_overlap_max_random (1)	Maximum-random overlap (in v13.x *cloud_overlap* = 1)
	cloud_overlap_2col_max_frac (2)	Two column, effective frac is maximum value among levels at pressures below *cc_low_cloud_top* option (in v13.x *cloud_overlap* = 2)
	cloud_overlap_2col_weighted (3)	Two column, effective frac is computed as hydrometeor-weighted average

		(v13.x RTTOV-SCATT default)
	cloud_overlap_2col_user_frac (4)	Two column, user specifies effective frac in *hydro_frac_eff* profile variable (replaces the *lusercfrac* option in RTTOV-SCATT v13.x).

The following options are new in v14 and are used for selecting features that were previously activated in different ways in v13 (see section 12 on scattering simulations):

- *opts % scatt % radar*
- *opts % scatt % baran_ice_version*
- *opts % cloud_overlap % per_hydro_frac*

The following options have been removed as they either became redundant in v14, were previously deprecated in v13, or otherwise are no longer required:

- *opts % config % fix_hgpl*
  - v14 behaviour equivalent to true in v13 (deprecated option in v13)
- *opts % interpolation % spacetop*
  - v14 behaviour equivalent to true in v13 (deprecated option in v13)
- *opts % interpolation % reg_limit_extrap / opts_scatt % reg_limit_extrap*
  - v14 behaviour equivalent to true in v13 (deprecated option in v13)
- *opts % rt_all % dtau_test / opts_scatt % dtau_test*
  - v14 behaviour equivalent to false in v13 (deprecated option in v13)
- *opts % rt_mw % fastem3_rwd_fix / opts_scatt % fastem3_rwd_fix*
  - FASTEM-3 is no longer implemented.
- *opts % rt_ir % rayleigh_depol*
  - v14 behaviour equivalent to true in v13, more accurate treatment of Rayleigh phase function
- *opts % rt_ir % grid_box_avg_cloud*
  - v14 behaviour equivalent to true in v13, hydrometeor concentrations are input as grid box average values which is consistent with NWP model fields

Finally, **it is important to take note of the default values of the options.** Some have changed since RTTOV v13. It is good practice to set all options explicitly that are not shaded in grey indicating "no change usually necessary" in Table 8.3 to be certain of the configuration.

- *opts % config % adk_bt* (*switchrad* in v13) to select input perturbation units for thermal channels for the AD/K models now applies to all simulations (including MW scattering) and defaults to true (i.e., AD/K in terms of BT), whereas in v13, the RTTOV default was false. See section 15.
- *opts % config % adk_refl* is a new option allowing selection of input perturbation units for UV/VIS/NIR channels for the AD/K models. It defaults to true meaning inputs should be in reflectance rather than radiance as in v13.
- *opts % interpolation % enable_interp* defaults to true and should not usually be set to false.
- the default *opts % interpolation % interp_mode* is mode 4 in v14 (Rochon interpolation with optical depth weighting function interpolation). See section 9.
- *opts % rt_all % use_t2m* replaces the *use_t2m_opdep* option in v13. It now enables/disables the use of the 2m temperature *everywhere* in the code, i.e., the 2m temperature is now an

optional input (whereas it was mandatory in v13). See section 7.4.1 of the RTTOV v14 user guide.

- the default *opts % surface % mw_sea_emis_model* (*fastem_version* in v13) to select the microwave sea surface emissivity model is SURFEM-Ocean instead of FASTEM-6 in v13. See section 11.
- the default *opts % scatt % thermal_solver* (*ir_scatt_model* in v13) to select the scattering solver for thermal radiation is delta-Eddington. See section 12.
- the default *opts % cloud_overlap % overlap_param* to select the cloud overlap parameterisation is “auto select” which uses maximum/random overlap for VIS/IR sensors and two-column hydrometeor-weighted overlap for MW sensors (the generally recommended options for each). See section 12.
- the new *opts % cloud_overlap % col_threshold* applies to all cloud overlap parameterisations and defaults to zero. This replaces *cc_threshold* in RTTOV-SCATT in v13 which defaulted to 0.001. See section 12.
- radar simulations are now done simultaneously with passive radiance simulations which means options must be set appropriately for the passive radiance calculations (solver, cloud overlap, etc) when doing radar simulations. See section 12.

9. Input profiles

The most significant change between v13 and v14 is the way that the input profile is represented. Section 7.4 in the RTTOV v14 user guide covers this in detail (compare with section 7.3 in the RTTOV v13 user guide). Figure 9.1 illustrates the input profiles in RTTOV and RTTOV-SCATT in v13, and the new profile structure in RTTOV v14.

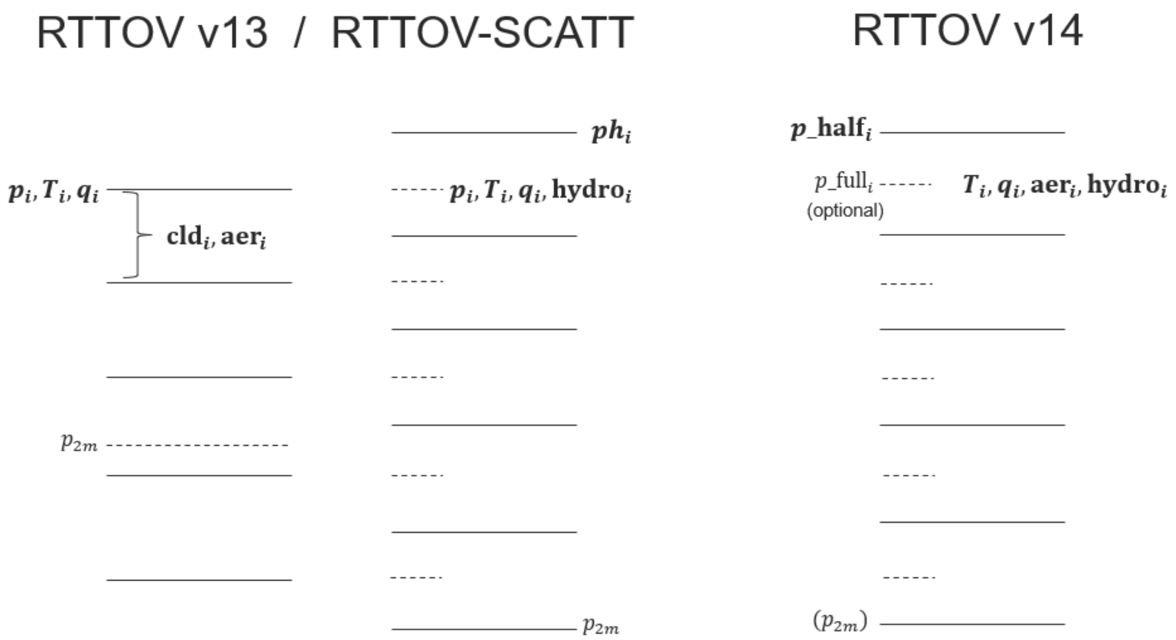

Figure 9.1: illustrating the input profile structures for RTTOV and RTTOV-SCATT in v13, and the unified profile structure in RTTOV v14.

RTTOV v14 requires pressure half-levels to be specified via *profiles(:)%p_half(1:nlevels)*: the half-levels bound the atmospheric layers in the vertical profile. In RTTOV v14 *nlevels* always refers to the number of half-levels. The top half-level is at the top of atmosphere and may be at 0 hPa. The bottom level is ***always*** at the surface. There is no longer a separate 2m (surface) pressure input to RTTOV.

The pressure half-levels define *nlayers=nlevels-1* atmospheric layers. Each layer is associated with a pressure full-level. You can choose to input these explicitly in *profiles(:)%p(1:nlayers)*, but if you do not specify them then RTTOV calculates each pressure full-level as the mean of adjacent half-level pressures. All other profile inputs (temperature, water vapour, trace gas concentrations, all scattering inputs) are provided on the pressure full-levels.

The implications of this change are that RTTOV v14 should always be run with the RTTOV interpolator enabled, and RTTOV v14 ***cannot*** be run directly on profile datasets specified on fixed pressure levels where the surface pressure is specified separately.

The benefits are that NWP model fields map more directly onto the profile inputs, and the previous inconsistency between the scattering inputs in the UV/VIS/IR and the MW is eliminated.

Section 7.4.1 of the RTTOV v14 user guide gives full details about the interpolation modes. Modes 4 or 5 are generally recommended as they generally result in the smallest interpolation errors among all modes. Mode 5 has the smallest errors overall while mode 4 gives smoother Jacobians.

Since there is no independent surface pressure profile variable, surface pressure Jacobians are obtained in *profiles_k(:)%p_half(nlevels)* only if *opts%interpolation%pressure_gradients* is set to true.

These changes in the profile levels/layers mean that it is not possible to replicate RTTOV v13 radiances in v14. The differences between v13 and v14 are documented in the RTTOV v14 Science and Validation Report. The RTTOV v14 profile structure is given in Table 9.1.

Table 9.1: RTTOV v14 profile structure.

Input profile members	Description	Units	Mandatory?	When used	Variable for TL?
nlevels	Number of pressure half-levels *Populated by allocation routine.*		Y		N
nlayers	Number of pressure full-levels/ atmospheric layers (i.e., *nlevels-1*) *Populated by allocation routine.*		Y		N
nsurfaces	Number of surfaces. *Populated by allocation routine.*		Y		N
gas_units	Units of gas abundances *(must be the same for all profiles)*	0,1,2	Y		N
p_half(:)	Pressure half-levels (*nlevels*)	hPa	Y		Y if *opts % interpolation % pressure_gradients* true

p(:)	Pressure full-levels (*nlayers*)	hPa	N	Always, computed internally if not specified.	Y if *opts % interpolation % pressure_gradients* true
t(:)	Temperatures on full-levels	K	Y		Y
q(:)	Water vapour concentration on full-levels	ppmv or kg/kg	Y		Y
o3(:)	O_3 concentration on full-levels	ppmv or kg/kg	N	If o3_data option .true.	Y
co2(:)	CO_2 concentration on full-levels	ppmv or kg/kg	N	If co2_data option .true.	Y
n2o(:)	N_2O concentration on full-levels	ppmv or kg/kg	N	If n2o_data option .true.	Y
co(:)	CO concentration on full-levels	ppmv or kg/kg	N	If co_data option .true.	Y
ch4(:)	CH_4 concentration on full-levels	ppmv or kg/kg	N	If ch4_data option .true.	Y
so2(:)	SO_2 concentration on full-levels	ppmv or kg/kg	N	If so2_data option .true.	Y
clw(:)	Cloud liquid water on full-levels treated as absorbing medium; not used in scattering simulations.	kg/kg	N	MW non-scattering only, if clw_data option .true.	Y
mmr_aer	Logical flag to set aerosol units: true => kg/kg, false => cm^{-3} *(must be the same for all profiles)*	T/F	N	Aerosol simulations with aertable file	N
mmr_hydro	Logical flag to set hydrometeor units: true => kg/kg, false => $g.m^{-3}$ *(must be the same for all profiles)*	T/F	N	Hydrometeor simulations with hydrotable file	N
flux_conversion(:) size (nhydro)	Use to specify units of flux for rain and snow hydrometeor types (*nhydro* determined by hydrometeor optical property file)	0,1,2	N	Optional with hydrometeor simulations with hydrotable file	N
aerosols(:,:) size (naer, nlayers)	Aerosol components/species concentration on full-levels (*naer* determined by aerosol optical property file)	kg/kg or cm^{-3}	N	Aerosol simulations with aertable file	Y
hydro(:,:) size (nhydro, nlayers) MW or (nhydro+1, nlayers) UV/VIS/IR	Hydrometeor concentration on full-levels (*nhydro* determined by hydrometeor optical property file)	kg/kg or $g.m^{-3}$	N	Hydrometeor simulations with optical property file	Y
hydro_frac(:,:) size (1, nlayers) or same as hydro(:,:) depending on per_hydro_frac option	Cloud/hydrometeor fractional cover on full-levels	0-1	N	Hydrometeor simulations	Y
hydro_frac_eff	Effective cloud/hydrometeor fraction for the whole profile	0-1	N	Hydrometeor simulations with cloud overlap parameterisation allowing user-specified *hydro_frac_eff*	Y
clwde_param	Cloud liquid water effective diameter parameterisation	1	N	UV/VIS/IR hydrometeor simulations with hydrotable file	N
icede_param	Ice effective diameter parameterisation	1,2,3,4	N	UV/VIS/IR hydrometeor simulations with hydrotable file	N

hydro_deff(:,:) *size (nhydro, nlayers)*	Hydrometeor effective diameter	microns	N	Optional with UV/VIS/IR hydrometeor simulations with hydrotable file	Y
surface_fraction(:) *size (nsurfaces-1)*	Fractional coverage of surfaces 1, …, *nsurfaces*-1. RTTOV automatically calculates fraction for the final surface.	0-1	N	Ignore for *nsurfaces=1* but must be specified for *nsurfaces>1.*	Y
near_surface(isurf) % t2m *1 <= isurf <= nsurfaces*	2m temperature	K	N	If *use_t2m* option is .true.	Y
near_surface(isurf) % q2m	2m water vapour	ppmv or kg/kg	N	If *use_q2m* option is .true.	Y
near_surface(isurf) % wind_u10m, *near_surface(isurf) % wind_v10m*	10m wind u, v components	m/s	N	Sea surface emissivity and BRDF models (except ISEM)	Y
near_surface(isurf) % wind_fetch	Wind fetch (length of water over which the wind has blown). Typical default: 100000 m	m	N	Sea surface BRDF model	Y
skin(isurf) % surftype *1 <= isurf <= nsurfaces*	Surface type (land = 0, sea = 1, sea-ice = 2)	0,1,2	Y	Emissivity and BRDF models and atlases	N
skin(isurf) % watertype	Water type (fresh = 0, ocean = 1)	0,1	N	Surface BRDF model and BRDF atlas	N
skin(isurf) % t	Surface skin temperature	K	N	Usually required, but not if *use_tskin_eff* option is true.	Y
skin(isurf) % salinity	Ocean salinity	Practical salinity unit	N	FASTEM-5/6, SURFEM-Ocean	Y
skin(isurf) % foam_fraction	Ocean foam fraction	0-1	N	FASTEM-5/6 if *use_foam_fraction* option .true.	Y
skin(isurf) % snow_fraction	Surface snow cover fraction	0-1	N	IR emissivity atlases	N
skin(isurf) % fastem(1:5)	FASTEM land/sea-ice parameters		N	FASTEM for land/sea-ice surface types	Y
zenangle	Satellite zenith angle	deg	Y		N
azangle	Satellite azimuth angle (0-360; measured clockwise, east=+90)	deg	N	Using FASTEM-5/6, SURFEM-Ocean, or solar option	N
sunzenangle	Solar zenith angle	deg	N	Solar option, NLTE option	N
sunazangle	Solar azimuth angle (0-360; measured clockwise, east=+90)	deg	N	Solar option	N
latitude	Latitude (-90 to +90)	deg	Y	For radiation path geometry and emissivity and BRDF atlases	N
longitude	Longitude (0-360)	deg	N	Emissivity and BRDF atlases	N
elevation	Surface elevation	km	Y	For radiation path geometry	N
cfraction	Cloud fraction for simple cloud	0-1	N	Simple cloud scheme only	Y
ctp	Cloud top pressure for simple cloud	hPa	N	Simple cloud scheme only (if *cfraction* > 0)	Y
be	Earth magnetic field strength	Gauss	N	Zeeman simulations	N
cosbk	Cosine of the angle between the Earth magnetic field and wave propagation direction		N	Zeeman simulations	N

date(1:3)	Date of the profile as year (e.g. 2024), month (1-12), and day (1-31)		N	Optional for solar calculations.	N
time(1:3)	Time of profile as hour, minute, second.		N	Not used by core RTTOV. Used by *rttov_calc_solar_angles* subroutine.	N

The following profile members are new in v14:

- *p_half(1:nlevels)* - pressure half levels bounding the atmospheric layers, with the surface at the bottom pressure half level. All other vertical profiles are provided on (1:*nlayers*) pressure full levels.
- *flux_conversion(1:nhydro)* - set to 1 (rain) or 2 (snow) for the corresponding hydrometeor types for inputs in flux units. All other values must be 0. Must be set explicitly for all members of the *profiles(:)* array. See section 12.
- *hydro_frac_eff* - optional explicit input of effective hydrometeor (cloud) fraction for two column overlap. See section 12.
- *nsurfaces* – number of surfaces (see section 11). As for *nlevels* and *nlayers,* this is populated by the allocation routine so there is no need to set it explicitly.
- *surface_fraction(:)* – fractional surface coverage for surfaces 1, 2, …, *nsurfaces*-1. RTTOV computes the fraction of the final surface so that the total coverage is 1. Ignore this if *nsurfaces=1.* See section 11.

The following v13 profile members have been renamed/replaced:

- *s2m%t/q/u/v/wfetc* – renamed *near_surface(:)%t2m/q2m/wind_u10m/wind_v10m/ wind_fetch*
- *cloud(:,:)* - renamed *hydro(:,:)*
- *cfrac(:,:)* - renamed *hydro_frac(:,:)*
- *clwde(:)* and *icede(:)* - replaced by *hydro_deff(:,:)*
- *mmr_cldaer* - replaced by *mmr_hydro* and *mmr_aer* now allowing independent choice of hydrometeor and aerosol input units.

The following v13 profile members have been removed:

- *s2m%p* - surface lies on the bottom pressure half-level so 2m/surface pressure is no longer an independent input variable.
- *s2m%o* - 2m ozone was never used.
- *skin%soil_moisture* - soil moisture was never used.
- *clw_scheme, ice_scheme* - UV/VIS/IR cloud simulations are now aligned with aerosol and microwave scattering in allowing arbitrary combinations of hydrometeor types so these profile variables specifying cloud liquid and ice water schemes are no longer required (see section 12).

For microwave scattering there is no separate “cloud profile” structure as for RTTOV-SCATT. The *hydro(:,:)* and *hydro_frac(:,:)* arrays have the **opposite index order** compared to the corresponding arrays in the *rttov_profile_cloud* structure in RTTOV-SCATT in v13.

The *near_surface* and *skin* members are now arrays of size *nsurfaces* allowing independent properties to be specified for multiple surfaces. See section 11.

Some profile variables are integer values and allow selection among different options or parameterisations. In RTTOV v13.x the integer values were sometimes explicitly used to select among the available choices. In RTTOV v14.0 integer constants are defined in the *rttov_const* module for all relevant profile variables (see section 5 above and Annex K of the v14 user guide), and it is recommended to use these in preference to the explicit integer values where possible. This makes code self-documenting and ensures that it will continue to work if the underlying integer values change in future RTTOV releases. Table 9.2 lists the integer constants available for each integer profile variable.

Table 9.2: named integer constants defined in rttov_const recommended to be used when setting integer profile variables, and their v13.x equivalents.

Profile variable	Integer constants defined in *rttov_const*	Description
profile % **gas_units**	gas_unit_ppmvdry (0)	Input gas profiles and 2m water vapour in units of ppmv over dry air (no change from v13.x)
	gas_unit_kg_per_kg (1)	Input gas profiles and 2m water vapour in units of kg/kg over moist air (no change from v13.x in value, but the constant name was previously *gas_unit_specconc*)
	gas_unit_ppmv (2)	Input gas profiles and 2m water vapour in units of ppmv over moist air (no change from v13.x)
profile % skin % **surftype**	surftype_land (0)	Land surface (no change from v13.x)
	surftype_sea (1)	Ocean or water surface (no change from v13.x)
	surftype_seaice (2)	Sea-ice surface (no change from v13.x)
profile % skin % **watertype**	watertype_fresh_water (0)	For *surftype_sea*, fresh water (no change from v13.x)
	watertype_ocean_water (1)	For *surftype_sea*, saline water (no change from v13.x)
profile % **clwde_param**	clwde_martin (1)	Martin et al CLW Deff parameterisation (no change from v13.x, but the constant is new)
profile % **icede_param**	icede_ou_liou (1)	Ou and Liou ice Deff parameterisation (no change from v13.x, but the constant is new)
	icede_wyser (2)	Wyser et al ice Deff parameterisation (no change from v13.x, but the constant is new)
	icede_boudala (3)	Boudala et al ice Deff parameterisation (no change from v13.x, but the constant is new)
	icede_mcfarquhar (4)	McFarquhar ice Deff parameterisation (no change from v13.x, but the constant is new)
profile % **flux_conversion(:)**	flux_conv_none (0)	Units for corresponding hydrometeor are determined by *profile % mmr_hydro* (no change from v13.x, but the constant is new)
	flux_conv_rain (1)	Units for corresponding hydrometeor are rain flux (kg/m^2/s) (no change from v13.x, but the constant is new)
	flux_conv_snow (2)	Units for corresponding hydrometeor are snow flux (kg/m^2/s) (no change from v13.x, but the constant is new)

10. Checking of input data

This is described in section 7.7 of the v14 user guide and the relevant subroutine interfaces are given in Annex I. For comparison, information on v13 is in section 7.3 of the v13 user guide.

RTTOV v13 provided the *rttov_user_options_checkinput* subroutine for checking consistency of options and coefficients before calling RTTOV. In v14 this subroutine is now called *rttov_user_check_options* and carries out additional checks, including optionally on the *chanprof(:)* argument to RTTOV.

RTTOV v13 provided the *rttov_user_profile_checkinput* subroutine for checking a single input profile for unphysical values, and optionally also to check it against the gas absorption optical depth regression limits, and the aerosol profile limits used in training PC-RTTOV aerosol coefficients. Optionally, this routine could be used to check explicit optical property input structures (*rttov_opt_param*).

In v14, this subroutine is replaced by three new routines. The *rttov_user_check_profile* subroutine performs the same checks on a single profile structure as the v13 routine above, with the addition of optionally checking against the hydrometeor concentration training limits for PC-RTTOV hydrometeor simulations (section 13), and additionally allows for fatal errors caused by unphysical profile values to be returned silently (i.e., no printed output). This latter new feature allows unphysical profiles to be screened out before calling RTTOV without generating unnecessary output.

Two new subroutines, *rttov_user_check_emis_refl* and *rttov_user_check_opt_param* can be used to check *rttov_emis_refl* and *rttov_opt_param* structures for invalid values.

By default, RTTOV checks these various input structures for unphysical values internally. The *opts%config%check_profiles* option (previously "*check_input*" in v13) can be set to false to turn off checking of the *profiles(:), emis_refl(:),* and *hydro/aer_opt_param* arguments. This can be done if you want to use the user-level routines described above to check the structures before calling RTTOV.

In v13, setting the *check_input* option to false would also turn off the checking of profiles against the regression limits. In v14, these checks are *always* done within RTTOV so that the *radiance%quality(:)* output always indicates when input values are restricted or clipped to some minimum or maximum value within RTTOV (see section 14).

11. Surface emissivity and reflectance

The main changes related to surface emissivity and reflectance are:

- all inputs/outputs related to emissivity and reflectance have been gathered into a single new derived type *rttov_emis_refl.*
- users have full control over the diffuse reflectance (the surface reflectance used for radiation emitted or scattered downward by the atmosphere) just as for emissivity and BRDF.
- a new heterogenous surface capability is available which provides improved simulations in the case of multiple surfaces (e.g., land and sea, or sea and sea-ice) falling within the satellite footprint.
- a reduced set of MW sea emissivity models are available.

- updates related to the emissivity and BRDF atlases.

Section 8.3 of the RTTOV v14 user guide provides full details of the treatment of surface emissivity and reflectance in RTTOV v14. For comparison the v13 user guide covers this in sections 7.5 (emissivity), 7.6 (BRDF), and 8.4 (additional information on emissivity and reflectance).

Inputs/outputs

In RTTOV v13, emissivity and reflectance inputs/outputs were separated into the *rttov_emissivity* and *rttov_reflectance* structures. Arrays of these types along with logical input arrays *calcemis(:)* and *calcrefl(:)* were defined by the user and input to the *rttov_direct* (and TL/AD/K) routine.

In RTTOV v14, these have been replaced by a single *rttov_emis_refl* structure which contains all members of the *rttov_emissivity* and *rttov_reflectance* structures and also *calc_emis(:)* and *calc_brdf(:)* members which correspond to the previous *calcemis* and *calcrefl* arrays. The previous *refl_in* and *refl_out* members of *rttov_reflectance* have been renamed *brdf_in* and *brdf_out*.

All members of the new structure are arrays, and an array of *rttov_emis_refl* structures (see below) can be (de)allocated via the *rttov_alloc_direct/tl/ad/k* routines (recommended) or via the new *rttov_alloc_emis_refl* subroutine. Only members that are required based on the selected options are allocated.

The new *rttov_emis_refl* data type is shown in Table 11.1 (all arrays are of size *nchanprof,* i.e., the same size as the *chanprof(:)* argument to RTTOV).

Table 11.1: the new rttov_emis_refl *data structure.*

Type	Variable	Description
Logical	calc_emis(:)	Where true, channel emissivities are provided by RTTOV. Where false, you provide input emissivity values in *emis_in(:)*. Only relevant for thermal channels.
Real	emis_in(:)	Input emissivities, used where *calc_emis(:)* is false.
Real	emis_out(:)	Emissivities used by RTTOV.
Logical	calc_diffuse_refl(:)	Where true, channel diffuse reflectances are provided by RTTOV. Where false, you provide input diffuse reflectance values in *diffuse_refl_in(:)*. Relevant for all channels.
Real	diffuse_refl_in(:)	Input diffuse reflectances, used where *calc_diffuse_refl(:)* is false.
Real	diffuse_refl_out(:)	Diffuse reflectances used by RTTOV.
Real	specularity(:)	Input surface specularity (0-1). 0 => fully Lambertian, 1 => fully specular. Only used if *opts%surface%lambertian* option is true.
Real	tskin_eff(:)	Input per-channel effective skin temperature (K). Only used if *opts%surface%use_tskin_eff* option is true. On output, always contains skin temperatures used by RTTOV.
Logical	calc_brdf(:)	Where true, channel BRDFs are provided by RTTOV. Where false, you provide input BRDF values in *brdf_in(:)*. Only relevant for solar channels when *opts%rt_all%solar* is true.
Real	brdf_in(:)	Input BRDFs, used where *calc_brdf(:)* is false.
Real	brdf_out(:)	BRDFs used by RTTOV.

Diffuse reflectance

The diffuse reflectance is used in RTTOV for radiation emitted or scattered downwards by the atmosphere and is relevant to all channels. There is a new *calc_diffuse_refl(:)* member which specifies whether RTTOV should compute the diffuse reflectance (true) or whether the user is supplying diffuse reflectance values (false) via the *diffuse_refl_in(:)* member. In all cases, the values used in the RTTOV calculations are output in the *diffuse_refl_out(:)* member. This is new behaviour: in v13 it was only possible to specify the diffuse reflectance for VIS/NIR channels when *calcrefl(:)* was false. Now it can be specified for *all* channels independently of *calc_emis(:)* and *calc_brdf(:)*. In v13 the diffuse reflectances were BRDF-like for VIS/NIR channels, but in v14 they are BRF-like. In other words, the input and output diffuse reflectance values for VIS/NIR channels in v14 are π times the v13 values.

In general it is recommended to set *calc_diffuse_refl(:)* to true for all channels unless you are confident in the values you wish to supply: by setting this to true the v13 behaviour is replicated. For thermal channels, RTTOV computes the diffuse reflectance as (1-*emissivity)* except in the case of MW sensors over sea surfaces where FASTEM or SURFEM-Ocean are used in which case the reflectance is modified to account for non-specular reflection. For VIS/NIR channels with *calc_diffuse_refl(:)* set to true, the diffuse reflectance is either computed from fixed water reflectance spectra for sea surfaces or otherwise is set to the BRDF multiplied by π.

In RTTOV v13, for non-thermal channels (at wavelengths below 3 µm), when the sea sun-glint model was used, a fixed surface reflectance taken from USGS water reflectance spectra was added to the computed sun-glint BRDF. These fixed reflectance spectra were also used for the diffuse reflectance. In v14, the diffuse reflectance (whether calculated by RTTOV or input by the user) is added to the sun-glint BRDF instead. This allows you to specify alternative diffuse reflectance values that may incorporate sub-surface scattering, and these will be accounted for in the BRDF as well.

Summary

Table 11.2 summarises the emissivity, BRDF, and diffuse reflectance values used under all conditions, and Table 11.3 summarises the corresponding TL inputs/calculations.

Table 11.2: summarising emissivity, BRDF, and diffuse reflectance in different spectral regions for each surface type.

<table>
<tr><th rowspan="2">Variable</th><th rowspan="2">calc_* Boolean</th><th rowspan="2">Surface type</th><th colspan="3">Variable values at channel wavelengths</th></tr>
<tr><th><3 µm</th><th>3-5 µm</th><th>>5 µm</th></tr>
<tr><td rowspan="4">Emissivity</td><td rowspan="3">calc_emis T</td><td>Sea</td><td rowspan="4">N/A</td><td colspan="2">IR: ISEM, IREMIS
MW: SURFEM-Ocean, FASTEM-5/6</td></tr>
<tr><td>Land</td><td colspan="2">IR: 0.98
MW: FASTEM land/sea-ice parameterisation</td></tr>
<tr><td>Sea-ice</td><td colspan="2">IR: 0.99
MW: FASTEM land/sea-ice parameterisation</td></tr>
<tr><td>calc_emis F</td><td>All</td><td colspan="2">User input emissivity
emis_refl%emis_in(:)</td></tr>
<tr><td rowspan="4">BRDF</td><td rowspan="3">calc_brdf T</td><td>Sea</td><td>Sunglint BRDF model + diffuse reflectance</td><td>Sunglint BRDF model</td><td rowspan="4">N/A</td></tr>
<tr><td>Land</td><td>0.3/π</td><td rowspan="2">(1-emissivity)/π</td></tr>
<tr><td>Sea-ice</td><td>0.8/π</td></tr>
<tr><td>calc_brdf F</td><td>All</td><td colspan="2">User input BRDF
emis_refl%brdf_in(:)</td></tr>
<tr><td rowspan="4">Diffuse reflectance</td><td rowspan="3">calc_diffuse_refl T</td><td>Sea</td><td>Fixed USGS spectra</td><td colspan="2">IR: (1-emissivity)
MW: computed by SURFEM-Ocean/FASTEM-5/6 if calc_emis T otherwise (1-emissivity)</td></tr>
<tr><td>Land</td><td rowspan="2">brdf*π</td><td colspan="2" rowspan="2">(1-emissivity)</td></tr>
<tr><td>Sea-ice</td></tr>
<tr><td>calc_diffuse_refl F</td><td>All</td><td colspan="3">User input diffuse reflectance
emis_refl%diffuse_refl_in(:)</td></tr>
</table>

Table 11.3: summarising the emissivity, BRDF, and diffuse reflectance TL inputs/calculations in each spectral region, and for each surface type. For most ocean surface models, where the ***calc_**** *variable is true, the corresponding TL variables are computed from the profile TL inputs for the variables used by the model such as 10m wind u/v components, skin temperature, salinity, etc. The exceptions are ISEM for emissivity and diffuse reflectance in VIS/NIR channels (<3 um) over sea.*

<table>
<tr><th rowspan="2">Variable</th><th rowspan="2">calc_* Boolean</th><th rowspan="2">Surface type</th><th colspan="3">$\delta\epsilon, \delta b, \delta r$ values at channel wavelengths</th></tr>
<tr><th><3 µm</th><th>3-5 µm</th><th>>5 µm</th></tr>
<tr><td rowspan="4">Emissivity TL $\delta\epsilon$</td><td rowspan="3">calc_emis T</td><td>Sea</td><td rowspan="4">N/A</td><td colspan="2">IR: ISEM: user input $\delta\epsilon$ (emis_refl_tl%emis_in)
IR: IREMIS: $\delta\epsilon$ computed
MW: SURFEM-Ocean, FASTEM-5/6 $\delta\epsilon$ computed</td></tr>
<tr><td>Land</td><td colspan="2">IR: user input $\delta\epsilon$ (emis_refl_tl%emis_in)
MW: FASTEM land/sea-ice param $\delta\epsilon$ computed</td></tr>
<tr><td>Sea-ice</td><td colspan="2">IR: user input $\delta\epsilon$ (emis_refl_tl%emis_in)
MW: FASTEM land/sea-ice param $\delta\epsilon$ computed</td></tr>
<tr><td>calc_emis F</td><td>All</td><td colspan="2">User input $\delta\epsilon$
emis_refl_tl%emis_in</td></tr>
<tr><td rowspan="3">BRDF TL δb</td><td rowspan="2">calc_brdf T</td><td>Sea</td><td>Sunglint BRDF model δb computed + δr</td><td>Sunglint BRDF model δb computed</td><td rowspan="3">N/A</td></tr>
<tr><td>Land
Sea-ice</td><td>User input δb
emis_refl_tl% brdf_in</td><td>$-\delta\epsilon/\pi$</td></tr>
<tr><td>calc_brdf F</td><td>All</td><td colspan="2">User input δb
emis_refl_tl%brdf_in</td></tr>
<tr><td rowspan="3">Diffuse reflectance TL δr</td><td rowspan="2">calc_diffuse_refl T</td><td>Sea</td><td>User input δr
emis_refl_tl% diffuse_refl_in</td><td colspan="2">IR: $-\delta\epsilon$
MW: $\delta\epsilon$ computed by SURFEM-Ocean/FASTEM-5/6 if calc_emis T otherwise $-\delta\epsilon$</td></tr>
<tr><td>Land
Sea-ice</td><td>$\delta b*\pi$</td><td colspan="2">$-\delta\epsilon$</td></tr>
<tr><td>calc_diffuse_refl F</td><td>All</td><td colspan="3">User input δr
emis_refl_tl% diffuse_refl_in</td></tr>
</table>

Heterogenous surface capability

RTTOV v13 assumed surfaces to be homogeneous with a single surface type (land, sea, or sea-ice) and a single set of associated properties. RTTOV v14 provides the ability to account for heterogenous surfaces by providing surface, skin, and near-surface variables for multiple surfaces. These are then combined to obtain the surface properties when solving the radiative transfer equation. All profile near-surface and skin variables are specified per surface, as are the contents of the *rttov_emis_refl* structure (input/output emissivities, reflectances, and associated data).

The interface changes to support this capability are:

- some allocation subroutines (*rttov_alloc_direct/tl/ad/k, rttov_alloc_prof, rttov_alloc_traj_all*) take an additional *nsurfaces* argument which is the number of surfaces per profile. For the homogeneous case, set this to 1. All profiles are associated with the same number of surfaces. Typically, *nsurfaces* will be 1 or 2, but any number of surfaces are allowed.
- the *rttov_profile* structure (see section 9) has a new *surface_fraction(1:nsurfaces-1)* member in which the coverage fraction for each surface is specified. The surface coverage fraction for

surfaces 1, …, *nsurfaces*-1 are specified here. RTTOV computes the final surface fraction as *1-SUM(surface_fraction(:))* to ensure the total coverage is 1. For the homogeneous case, this profile member should be ignored as the fraction is implicitly 1.

- the *near_surface* and *skin* members of the *rttov_profile* structure are now arrays of size *nsurfaces*. Each member must be populated with the relevant variables for each defined surface. For the homogeneous case, set the members of *profiles(:)%near_surface(1)* and *profiles(:)%skin(1).*
- the *emis_refl* argument to *rttov_direct/tl/ad/k* is an array of size *nsurfaces*. All relevant members should be populated for each surface. For the homogeneous case, *emis_refl* should be an array of size 1, and the relevant members of *emis_refl(1)* should be populated.
- when accessing the emissivity and BRDF atlases, a surface index is specified to identify which surface in the profile structure should be used when obtaining the emissivity/BRDF values from the atlas.

MW sea surface emissivity models

The MW sea surface emissivity model is selected via the *opts%surface%mw_sea_emis_model* option (*fastem_version* in v13). Only three of the options implemented in v13 remain available in v14: FASTEM-5, FASTEM-6, and SURFEM-Ocean, the last of which is now the default and is generally recommended.

Emissivity/BRDF atlases

The atlas files that were previously in HDF5 format are now in netCDF format and the relevant files must be downloaded for use with RTTOV v14. All existing atlases are supported and reproduce the outputs as in v13. In addition, RTTOV v14 supports the CAMEL v3 atlases which are recommended over the pre-existing v2 atlases. The atlas data files are available to download from the NWP SAF website except for the CAMEL v3 single-year atlas data: these are available for each month in the years 2000-2021 and must be downloaded directly from the NASA LP DAAC website (link available on the NWP SAF website).

For the IR emissivity and BRDF atlases, there is a new optional *max_distance* argument. If a positive value is supplied for this argument (a distance in km) and the atlas has no emissivity/BRDF data at the given lat/lon location, then a search is made and a valid emissivity/BRDF is returned if one is found within the distance *max_distance* of the original location. Note that it is not necessarily the nearest valid value: the search is optimised for speed. Set *max_distance* to zero to turn off this search.

12. Scattering simulations

Section 8.4 of the v14 user guide gives full details of scattering simulations. For comparison, the v13 user guide covers this in sections 8.5 (VIS/IR cloud simulations), 8.6 (aerosol simulations), and 8.7 (RTTOV-SCATT for MW scattering).

RTTOV v14 unifies the distinct RTTOV-SCATT MW scattering model in v13 with the core RTTOV model. This means that scattering simulations in v14 are carried out in essentially the same way for sensors at all wavelengths from a technical perspective. This section provides an overview of the scattering capabilities in v14.

Enabling scattering simulations

Hydrometeor scattering is enabled by setting *opts%scatt%hydrometeors* to true, and aerosol scattering by setting *opts%scatt%aerosols* to true. The scattering solvers for thermal (emitted) and solar radiation are selected via *opts%scatt%thermal_solver* and *opts%scatt%solar_solver* respectively. Section 8 gives the constants and values used to select the available options.

Thermal solvers

The delta-Eddington solver can be used for IR aerosol and hydrometeor, and MW hydrometeor simulations, with all cloud overlap options. It is the default *thermal_solver* because it is the fast option that supports all simulation types.

The Chou-scaling solver remains available for IR sensors but does not apply in the MW. The Tang *et al* (2018) modification has been implemented which can improve the accuracy of the Chou-scaling approximation for ice clouds in the far-IR. See the RTTOV v14 user guide for more information.

Solar solvers

The DOM solar solver remains available and is now the default option. In addition, the MFASIS-NN fast parameterisation is available. This has undergone improvements since v13 and now supports additional channels.

Radar simulations

Radar simulations are enabled by setting both *opts%scatt%hydrometeors* and *opts%scatt%radar* to true and by supplying the *reflectivity* argument to *rttov_direct* (and associated *reflectivity_tl/_ad/_k* arguments to the TL/AD/K interfaces). Unlike in RTTOV v13, passive radiances are output alongside the radar reflectivity profiles.

Cloud overlap parameterisations

The choice of cloud overlap is made in *opts%cloud_overlap%overlap_param*. Section 8 gives the constants and values used to select the available options.

Four overlap parameterisations are implemented, the two implemented in RTTOV v13 and the two implemented in RTTOV-SCATT. These may all be used with any solver. Maximum/random overlap is recommended for VIS/IR sensors, and the two-column hydrometeor-weighted option is recommended for MW sensors: these are selected when *overlap_param* is set to the “auto select” value (the default). There are three options for two-column schemes yielding one clear column and one cloudy column with an associated effective hydro fraction. The “simple cloud overlap” in v13 is now referred to as “two-column max frac” as it takes the maximum hydro fraction among the layers between the TOA and a user-specified maximum pressure. The RTTOV-SCATT hydrometeor-weighted two-column scheme is implemented, as is the option for two columns with a user-specified effective hydro fraction.

Cloud overlap parameterisations do not apply to the radar solver.

The *opts%cloud_overlap%col_threshold* option applies to all cloud overlap schemes and can be used to ignore any cloud columns with weights less than the specified value as in v13. The corresponding weights are added to the clear column weight. For two-column overlap schemes, if the effective hydro fraction is less than this threshold then the clear radiance is returned.

By default, a single profile of hydrometeor fractions is supplied in *profile%hydro_frac(1,:)*. The *per_hydro_frac* option previously implemented in RTTOV-SCATT is available, allowing separate hydro fractions for each hydrometeor. In this case the *hydro_frac(:,:)* array has the same dimensions as the *hydro(:,:)* array (see below). This can only be used with the two-column hydrometeor-weighted cloud overlap scheme. It also affects the radar solver calculations if activated.

The *hydro_frac_tlad* option implemented in RTTOV-SCATT is available which enables/disables hydrometeor TL/AD sensitivity to the hydro fraction. This only applies to the two-column hydrometeor-weighted cloud overlap scheme.

Polarisation

The empirical and ARO-scaling treatments of polarisation due to frozen hydrometeors for MW sensors implemented in RTTOV-SCATT are available in v14. The polarisation option is selected via the *opts%scatt%mw_pol_mode* option. The empirical scale factor is specified in *opts%scatt%ice_polarisation*. For the ARO-scaling option, the additional sensor-independent lookup table is read in with other coefficient files using the *rttov_read_coefs* subroutine.

Other scattering changes

Any non-zero scattering inputs (hydrometeors, aerosols, explicit optical properties) in the top-most layer are silently ignored by RTTOV v14. This is because the first pressure half-level (the upper boundary of this layer) can be at 0 hPa, in which case the thickness of this first layer is not well-defined.

The *zero_hydro_tlad* option implemented in RTTOV-SCATT (enabling hydrometeor concentration TL/AD/K sensitivity in clear layers when true) is implemented in v14, but currently only applies to MW sensors, and only with two-column overlap schemes and the delta-Eddington solver.

Pre-defined optical properties – hydrometeors

The hydrometeor inputs have been unified. This means that the UV/VIS/IR cloud inputs have been modified to be fully flexible, as for the aerosol and MW hydrometeor inputs. There is no longer any need to choose one liquid cloud scheme and one ice cloud scheme for optical properties. All optical property files (aerosol and hydrometeor, at all wavelengths) define some number of particle types, and these may be used individually or in any combination by assigning concentration profiles to the corresponding indices in the input arrays. This was already the case for aerosols, and this was introduced for RTTOV-SCATT hydrometeors in v13. RTTOV v14 now updates UV/VIS/IR cloud in the same way.

The most significant difference between the UV/VIS/IR and the MW is that optical properties for different sets of hydrometeors are currently provided: these are currently the same hydrometeor/cloud types as in v13. In the future, the goal is to offer unified hydrometeor optical properties across the whole spectrum: this was not previously possible in v13. The updates in v14 also allow for the creation of user tools to generate custom hydrometeor properties at all wavelengths which will be considered for a future release.

In the UV/VIS/IR there are five OPAC liquid water types, the "Deff" cloud liquid type, the Baum ice properties, and the Baran ice scheme. The Baran scheme is a special case because it is fully parameterised in the code while all other properties are stored in the hydrometeor optical property files. These are the same optical properties previously available in v13.

In the MW, properties are provided for the same five hydrometeors as in v13 (rain, snow, graupel, liquid cloud, ice cloud).

Hydrometeor concentrations for all sensors are input via the *profile%hydro(:,1:nlayers)* array. In the MW the first dimension has size *nhydro,* the number of particle types defined in the optical property file (currently *nhydro=5*). For the UV/VIS/IR, the first dimension of *hydro(:,:)* is sized *nhydro+1.* Here *nhydro=7* corresponding to the five OPAC, Deff CLW, and Baum ice types. The additional "+1" corresponds to the Baran ice scheme. The required optical properties are used by assigning hydrometeor concentrations to the corresponding index. The choice among implemented Baran scheme versions (currently UV/VIS/IR-only) is made via the *opts%scatt%baran_ice_version* option.

Note that for MFASIS-NN simulations, you must ensure that hydrometeor profiles are assigned to the correct elements of the profile arrays corresponding to the optical properties compatible with the NN training (currently all except the Baran ice type). Any cloud data in other elements of the cloud arrays are ignored. Currently this means that cloud assigned to the Baran ice index is silently ignored by MFASIS.

Hydrometeor concentrations must always be grid box average values in v14: NWP model cloud fields typically provide this.

For UV/VIS/IR hydrometeor types that are parameterised in terms of particle size, the particle sizes may be specified explicitly in the *profile%hydro_deff(1:nhydro,1:nlayers)* array by assigning values to the indices corresponding to the particle types in question. This replaces and generalises the v13 *clwde(:)* and *icede(:)* profile variables. Where these values are less than or equal to zero in elements corresponding to non-zero hydrometeor concentrations, the RTTOV Deff parameterisations are used as in v13. The Baran scheme has no explicit dependence on particle size, so the *hydro_deff(:,:)* array does not have the additional "+1" in the size of the first dimension.

The *clwde_param* and *icede_param* scalar profile variables are used to select the liquid and ice cloud Deff parameterisations. This means that only one Deff parameterisation is applied to all Deff-dependent liquid clouds and one to all Deff-dependent ice clouds for each profile. For the current hydrometeor optical property files, there is only one of each (the CLW Deff type, and the Baum ice type).

The *profile%mmr_hydro* selects the hydrometeor units for all sensors: kg/kg if true, otherwise g/m^3. (Note that this is now independent of the aerosol units, unlike in v13). The *profile%flux_conversion(1:nhydro)* works the same as in v13 to select rain/snow flux (kg/m^2/s) inputs overriding *mmr_hydro*: assign 1 (rain) or 2 (snow) to the indices corresponding to rain/snow types to enable flux inputs, and set all other elements to zero. Note however that in v13 this was specified once when allocating the RTTOV-SCATT cloud profile structure while in v14 you must set it explicitly for every individual profile.

Pre-defined optical properties – aerosols

The aerosol optical property files available for v14 define properties for the same sets of aerosols as in v13. Aerosol profiles are assigned to *profile%aerosols(1:near,1:nlayers)*, where the number of aerosols, *naer,* is determined by the aerosol optical property file.

The *profile%mmr_aer* selects the aerosol units: kg/kg if true, otherwise number density cm^{-3}. (Note that this is now independent of the hydrometeor units, unlike in v13).

Explicit optical property inputs

Explicit optical property inputs work across the full spectrum for all passive solvers. The members of the *rttov_opt_param* structure have changed and are as follows:

- *ext(:,:)* - total extinction coefficient (km^{-1}), dimensions (*nlayers, nchanprof*), always required.
- *ssa(:,:)* - single scattering albedo [0,1], dimensions (*nlayers, nchanprof*), always required.
- *bpr(:,:)* - "*b* parameter" [0,1], dimensions (*nlayers, nchanprof*), Chou-scaling only.
- *asym(:,:)* - asymmetry parameter [-1,1], dimensions (*nlayers, nchanprof*), delta-Eddington only.
- *nmom* - number of Legendre coefficients specified for all phase functions, DOM solver only.
- *lcoef(:,:,:)* - coefficients of Legendre decomposition of phase function, dimensions (*1:nmom+1,nlayers, nchanprof*), DOM solver only (thermal and/or solar).
- *phangle(:)* - the angles over which the phase functions are defined (units: degrees), dimension (*nphangle*), solar DOM solver only.
- *pha(:,:,:)* - azimuthally-averaged phase function, dimensions (*nphangle, nlayers, nchanprof*), solar DOM solver only.

Note that the *abs* and *sca* members (absorption and scattering coefficients) have been replaced by *ext* and *ssa*, *legcoef* has been renamed *lcoef,* and a new member *asym* has been added for the asymmetry parameter (required for the delta-Eddington solver).

The *rttov_init_opt_param* subroutine is now only used to initialise an *rttov_opt_param* structure to zero, e.g., for AD/K simulations. The new *rttov_init_opt_param_solar* subroutine must be called for the direct model *rttov_opt_param* structure(s) when doing solar simulations in order to pre-calculate data for phase function interpolation. In v13, the *rttov_init_opt_param* subroutine fulfilled both functions.

Explicit hydrometeor optical property inputs must be provided for the cloudy fraction of each layer. There is a non-linear relationship between cloud amount and the optical properties which means they cannot be provided as grid box average values (unless the cloud fraction is one). This means the optical property inputs cannot be used with any overlap scheme that computes cloud fractions within RTTOV. Currently the hydrometeor property inputs are only supported with the max/random overlap.

The radar solver is not currently supported with explicit optical property inputs.

The polarisation scaling (empirical or ARO scaling) is not applied to explicit optical property inputs for MW sensors.

Finally, the TL/AD/K no longer have sensitivity to the Legendre coefficient (*lcoef*) and phase function (*pha*). All other variables (*ext, ssa, bpr, asym*) are active in the TL/AD/K.

13. PC-RTTOV

PC-RTTOV simulations in v14 are run in a very similar way to v13. The primary difference is that new PC coefficients must be downloaded. RTTOV v14 PC-RTTOV coefficients are available for IASI, IASI-NG and MTG-IRS at the time of release, and these must be used with the corresponding v13 predictor 7gas optical depth coefficient file. Each coefficient file supports all PC-RTTOV functionality (unlike in v13). This means simulations over any surface type, with any variable trace gas (excluding SO2 currently), and optionally with either NLTE, aerosol, or hydrometeor simulations. These last three options are mutually exclusive due to the way the PC coefficients are trained.

PC-RTTOV is now trained using the IREMIS sea surface emissivity model (so that it is consistent with standard RTTOV simulations, this is used automatically for PC-RTTOV simulations for sea surfaces) and using the CAMEL v3 climatology atlas for land surfaces (so it is recommended to use this atlas to provide land surface emissivities when calling PC-RTTOV).

The subset of channels that must be simulated can be obtained at run-time using the *rttov_get_pc_predictindex* subroutine: the interface to this routine has changed since v13 (see Annex I of the user guide).

The *opts%config%apply_reg_limits* option no longer applies to PC-RTTOV simulations: input profiles for PC simulations are never modified. Cases where input profiles exceed the training limits are still notified via printed warnings (if the *verbose* option is true) and via the *radiance%quality(:)* output (section 14).

Full details of PC-RTTOV simulations in RTTOV v14 are given in section 8.6 of the user guide.

14. Outputs

Section 7.8 of the v14 user guide gives details on RTTOV outputs.

Simulated radiances differ to v13 as documented in the RTTOV v14 Science and Validation Report.

The *radiance%quality(:)* output array operates in the same way as in v13, providing a bit mask that indicates where input values provided by the user have exceeded minimum/maximum values applicable to various parameterisations within RTTOV. The range of situations in which flags are set has been expanded in v14, except for MFASIS-NN which has fewer restrictions compared to the older MFASIS-LUT approach. Section 7.8.3 in the user guide provides full details on this.

A new option in v14, *opts%config%bt_overcast_calc,* may be set to true to enable additional output of brightness temperature equivalents of the overcast radiances in *radiance%bt_overcast(:,:)*.

A new derived type *rttov_diagnostic_output* contains the geometric heights of pressure half- and full-levels and the diagnostic effective hydro fraction (generalised for all cloud overlap schemes to be one minus the clear column fraction). These arrays are per-profile, whereas in v13 the geometric heights were output per-channel in size for consistency with other radiance outputs. There is now a new optional *diag_output* argument to *rttov_direct/tl/ad/k:* simply allocate and pass this argument to obtain these outputs.

In general, outputs in v13 that were on *nlevels* or *nlayers* remain on *nlevels* or *nlayers* (respectively) in v14, where each is one larger than in v13 for typical applications. The exceptions are geometric heights and radar reflectivities. The *diag_output%geometric_height(:,:)* contains the calculated altitudes of the full pressure levels *p(1:nlayers).* A new *diag_output%geometric_height_half(:,:)* array contains the altitudes of the pressure half-levels *p_half(2:nlevels)* (excluding the top level since it may be at 0 hPa). For radar simulations the reflectivities *zef(:,:)* and *azef(:,:)* are computed on the pressure full-levels *p(1:nlayers)*, and so *geometric_height(:,:)* contains the corresponding altitudes.

The *radiance%plane_parallel* output flag no longer exists. Previously this indicated whether strictly plane parallel simulations were performed, but this is no longer clear-cut due to the way the various solvers work.

In v13, RTTOV-SCATT did not expose the *transmission* output from RTTOV. This is now available as an output for all simulations.

The contents of *radiance%cloudy(:)* have changed: this was not previously set by RTTOV-SCATT, and for VIS/IR scattering simulations (including aerosol-only) it was set equal to *radiance%total(:)*. Now the *cloudy(:)* radiance contains the total radiance from all cloudy columns as if there were no clear column for all solvers. For two-column overlap schemes, this is equivalent to the radiance for the cloudy column. The *cloudy(:)* radiance is zero for aerosol-only scattering simulations.

In the *radiance2* output structure, the *surf* member gives the Planck radiance corresponding to the atmospheric temperature at each half-level bounding the bottom of each layer. The final element was previously overwritten with the Planck radiance corresponding to the skin temperature, but as a result of the heterogenous surface update this is now the surface-leaving radiance incorporating the emissivity. For the homogeneous surface case, this is equal to the Planck radiance of the skin temperature multiplied by the surface emissivity. With the *lambertian* option set to true, in the general case of heterogenous surfaces with different specularities, then there is no longer a simple relationship between the *dnclear* and *refldnclear* members of *radiance2* because the downwelling radiances and the surface-reflected radiances are computed as sums over the different surfaces with corresponding weights (see section 8.3.10 in the RTTOV v14 user guide).

The optional feature in RTTOV-SCATT to populate a structure with data that can be used to perform dynamic emissivity retrievals given corresponding observed brightness temperatures has been implemented in v14 and generalised. This now applies to clear-sky simulations, and scattering simulations with the Chou-scaling and delta-Eddington solvers. It is activated by passing the *emis_retrieval_terms* argument of type *rttov_emis_retrieval_terms* to the *rttov_direct* subroutine. The emissivity retrieval (and optionally also skin temperature retrieval) can then be performed by calling the *rttov_emissivity_retrieval* subroutine. This feature is not compatible with the DOM solver and it requires that *nsurfaces=1* (see section 11).

15. TL/AD/K models

This section summarises the differences between v13 and v14 related to the TL, AD and K models. Full details of calling the TL/AD/K models is given in section 7.9 of the v14 user guide (and see section 7.9 of the RTTOV v13 user guide for comparison).

The *opts%interpolation%pressure_gradients* option can be set to true to enable the profile *p_half(:)* and, if specified in the direct model profiles, *p(:)* members as active TL/AD/K variables.

For thermal channels, the *switchrad* option has been renamed *opts%config%adk_bt* for greater clarity: it determines whether input AD/K perturbations are in terms of radiance (if false) or BT (if true, now the default). Input perturbations are provided in the *radiance%total(:)* or *radiance%bt(:)* respectively as in v13. Note that the default value has changed to true.

For solar simulations, the Jacobians for non-thermal channels (at wavelengths less than 3 µm) may now be in radiance or reflectance according to the new *opts%config%adk_refl* option. If true (the default), the AD/K reflectance perturbations should be provided in *radiance%refl(:)* for the corresponding channels, otherwise the AD/K radiance perturbations should be provided in *radiance%total(:)*. There is no problem providing inputs in both reflectance and radiance arrays: RTTOV uses the relevant input according to the *adk_refl* option.

For scattering simulations with explicit optical properties, the Legendre coefficients (*lcoef*) and phase function (*pha*) members are no longer active TL/AD/K variables. All other optical properties (*ext, ssa, bpr, asym*) are active in the TL/AD/K models.

When performing radar simulations, the input AD perturbations may be supplied in radiance and/or reflectivity. For the K model, you should supply K perturbations for either radiance or for reflectivity (not both). In the case of reflectivities, the input perturbations should be supplied for all channels in only one layer at a time. To obtain the full radar reflectivity Jacobian matrix involves calling the RTTOV K model multiple times, perturbing reflectivities in each scattering layer one at a time. The user guide provides more details in section 8.4.4.

16. Wrapper

The Python (*pyrttov*) and C++ wrappers have been fully updated for all new developments in RTTOV v14 and work for all previously supported simulation types.

The wrapper interface is fully documented in *rttov-wrapper.pdf* in the *docs/* directory.

17. Test suite and example code

The test suite is fully up to date with developments. Test suite profiles and test definitions have been updated to be relevant to RTTOV v14, to ensure full code coverage, and to remove redundant cases.

The test suite is documented in *rttov-test.pdf* in the *docs/* directory but note that you do not have to read this in order to use RTTOV.

As described in section 5.4.2 of the user guide, an updated selection of example programs is included in *src/test/* which can form a basis for your application:

- **example_fwd.F90** – simple example for clear-sky direct model simulations
- **example_k.F90** – simple example calling K model for clear-sky simulations.
- **example_atlas_fwd.F90** – same as example_fwd.F90, but demonstrates use of emissivity and BRDF atlases
- **example_aer_file_fwd.F90** – aerosol scattering simulations using an aertable optical property file.
- **example_aer_param_fwd.F90** – aerosol scattering simulations passing explicit optical properties.
- **example_hydro_visir_file_fwd.F90** –VIS/IR hydrometeor scattering using a hydrotable optical property file.
- **example_hydro_mw_file_fwd.F90** – MW hydrometeor scattering using a hydrotable optical property file.
- **example_hydro_param_fwd.F90** – hydrometeor scattering simulations passing explicit optical properties.
- **example_hydro_mfasis_nn_fwd.F90** – visible MFASIS-NN hydrometeor scattering simulation.
- **example_pc_fwd.F90** – demonstrates calling PC-RTTOV

-- END --