

## Version 0.10: Aug. 23, 2009

Now, released under GNU GPL version 3.

### 1. Rewritten in Fortran 90/95 with a modern coding style.

This is phase I of our attempt to renew our coding style. Only simple code conversions and modifications were applied in this phase. In the next version (v1.0), phase II will complete the attempt.

- Now, MCARaTS would be compatible with any Fortran 90/95 compiler. GFortran or G95 can be used.
- Only a few common but non-standard Fortran 90/95 procedures are used: getarg, iargc, and erf, all of which can be used with usual Fortran compiler.
- All codes are in some specific module.
- Added "implicit none" at the top of every module. So that, all variables are explicitly declared.
- All major arrays are now allocatable.
- Real types of all floating point variables and values are encapsulated in global constants (e.g., R\_ and RD\_), so that there is no system-dependence of the codes. In addition, the user can easily modify the precisions of ALL variables and values.
- All procedure arguments are declared with intent(in/out/inout).
- Many other modifications for new coding style: no common block, no "goto" statement, and so on.

### 2. Improved user interface

- Binary format files for input/output can be used. Text format is also acceptable in this version. However, note that in the future, only binary format will be accepted.
- When data files are in binary format, there is possibility of data skipping and quick, random access.
- Namelist variables were renamed, and the format of the namelist input file was completely reorganized.
- Multiple calculation jobs in a single experiment.

### 3. Enhanced calculations of pathlength distribution statistics for radiances

- Calculation of pathlength statistics are possible for any kind of radiance.
- Three kinds of pathlength statistics are implemented as options.
  - Layer air mass factors: Calculation of layer-by-layer air mass factors could be applied for each radiance pixel.
  - Column air mass factors: Multiple density profiles (or "weighting function") could be given and used for calculations of multiple column-averaged air mass factors.
  - Time resolved histogram of radiance contribution is another option.

### 4. Volume rendering as a 3-D visualization tool

- Visualization of 3-D volume data (e.g., extinction coefficients) is possible. At present, five rendering methods with different qualities are implemented. Approximate, quick RTE solvers are available for this purpose.
- A few useful utilities for quick image generation are bundled in the package.

### 5. Flexible specification methods for various phase functions

- Non-digital kind index of phase function: Phase function can be interpolated for a point between [ipf, ipf+1], where ipf is phase function index.
- Phase function can be isotropic, Rayleigh, Henyey-Greenstein, or a function interpolated from user-specified database.

### 6. Improved camera model for radiance calculations

- Three angles to specify the camera geometry: zenith angle, azimuth angle, and image rotation angle.
- Two pixel mapping methods: rectangular or polar mapping.
- Free location of radiance sampling point
- Finite size of detector surface area: In previous versions, camera focus point was infinitesimal.

### 7. Revised several utility tools for general binary/text data processing

- Several useful tools for data processing are bundled in the package. They can be also used to process output data from the mcarats code. The output data is now always in binary format that is compatible with GrADS.

### 8. Exponential sub-layer profile for Planck function

- Sub-layer profile of Planck function is assumed to be exponential:  $\ln(B)$  is linear to altitude, where B is Planck function. This assumption would be better than linear or polynomial profile.

## 9. Removed minor features for simplicity

The followings were removed but could be added again in the future (by someone).

- Treatment of pseudo-sphericity and atmospheric refraction
- Recalculation feature of the mcarats executable by adding photons

## 10. Minor improvements

- Radiance of the 3rd kind: Angular distribution of radiance averaged over solid angle and area of horizontal plane at some user-specified layer interface can be calculated.
- Improved algorithm for phase function truncation, for higher accuracy
- MPI broadcast for input data, to reduce total file access.
- Exact calculation of source energy fractions for atmosphere and surface, even when they have different number of elements in the horizontal dimensions. Random locations of source photons are also determined in unbiased way.
- Gaussian quadrature is used for faster calculations, in cases that exact ray tracing is too time-consuming due to large vertical-to-horizontal aspect ratio of atmospheric cells or because the ray is too oblique.
- Only-one simulation target (flux, radiance, or volume rendering) in a single experiment: There would be quite rare cases that the user tries to get fluxes and radiances simultaneously.
- Improved local source model.
- Radiances averaged with weighting  $\cos Q$  ( $Q$  is angle from the camera image center direction) or no weighting can be calculated.
- Input extinction and absorption coefficients can be easily scaled by respective factors. These are useful for simple adjustments of input data.

[Edit](#)

## v0.9.5 : Mar. 1, 2009

This is a moderate upgrade from the previous version and the final version with legacy Fortran 77 coding style. Please note that a release of the next version (v0.10) is coming soon (probably in a few months). Therefore, urgent upgrade to v0.9.5 is not highly recommended except in the case that it is really needed.

### 1. Adaptive super-cell formation

- added for efficient maximum cross section method: merging cells to form a super-cell until inhomogeneity within the super-cell exceeds a threshold or the maximum extinction coefficient exceeds a limit.

### 2. LSRT BRDF

- Added Li-Sparse-Ross-Thick (LSRT) BRDF model that is a parameterized land BRDF model.

### 3. Improved numerical efficiency for camera radiances

- Significant increase of numerical efficiency for computing camera-like image (computer graphics) composed of angle-averaged local radiances, by adding an improved collision-forcing method, a new scaling method for local estimates, and an improved denoising technique.

### 4. Flexible numbers of surface elements

- The numbers of surface elements (pixels) can be freely specified being independent of the atmosphere. This adds a freedom for user, for example, to give a uniform properties for the entire domain or higher spatial resolution for surface than the atmosphere.

### 5. The 3rd medium with vertical profile

- The third medium can be specified by ext3rd, omg3rd, jpf3rd. These properties are assumed to be horizontally homogeneous. For example, the user can give Rayleigh scattering and two kinds of aerosol mixtures for the first, second, and third media, respectively.

### 6. Minor changes

- Within the code, the atmospheric layers are categorized in horizontally homogeneous and inhomogeneous layers.
- Removed a calculation method using analytical transmittance of photon weight for radiative flux and heating rate sampling.
- Removed the delta-isotropic approximation of phase function, because the method results in low accuracy and the other truncation approximation method is enough.
- For interpolating phase function,  $\log(P)$  is interpolated by a cubic spline.
- Revised the initialization algorithms for random number generator. Now, no machine architecture dependence exists.
- Scattering direction for Rayleigh scattering is determined by using the rejection method.
- Created artviewclip.f, which clips view domain for efficient sampling of local radiative quantities.
- The user can specify the minimum, medium, and maximum distance for sampling of local radiative quantities.
- New post-processing tool, psample.f, which trims and/or resample 2-D data from the mcarats output file.
- Changed the definition of ADF in the code, for comprehensive coding. Now the ADF is defined as PDF of scattering/emission direction so that the unit of it is  $1/\text{steradian}$ , instead of  $\text{PDF}/|\cos(q_1)|$ .

Definition of adfsrc differs for point source and solar source, respectively. See artsim and r\_artangdistr.

- The input parameter fsrc is now simply the spectral power (W/micron) of the source for localized source (jdefsrc=0).

#### 7. Bug fixes

- Fixed incorrect method for determination of random emission direction from local source. The direction is now determined so as to be isotropic within the angular cone with isotropic angular distribution, instead of Lambertian distribution on a plane normal to the emission center direction. By this bug fix, wide angular emission from a point can be correctly treated. In the previous version, only narrow beam could be treated correctly. Input parameter fsrc is now the emitted spectral power, instead of the power projected to the plane normal to the emission center direction.
- Fixed the shadowing factor formula, by replacing a factor  $1/v_i^{**2}$  with  $1/v_i$ .

[Edit](#)

## v0.9.1: Nov. 1, 2008

This is a bug-fix release. Changes from from v0.9 are as follows:

[Bugfixes]

- r\_dsmfshad.f: Fixed the incorrect formula of the shadowing factor. In the previous version, this bug could affect the BRDF for large viewing/solar zenith angles, for the rough micro-facet model.
- artnetemit.f: Fixed the sampling of heating rate for thermal radiation at the initial cell of photon emission. In the previous version, heating rate of thermal IR may be biased, especially at strong absorption bands.
- artreduce.f: Fixed the instruction of MPI reduce for a variable, pplen (path length statistics). This bug did not affect when  $n_{xr}=n_{yr}=1$ . There has been no report from user who calculates the air mass factors using MPI parallelization for 3-D inhomogeneous atmosphere with  $n_{xr}$  and  $n_{yr}$  larger than 1.

There are a few known issues listed below. They will be fixed soon in subsequent versions.

[Known issues]

- Treatment of wide beam source from a point is not accurate. Narrow beam is acceptable although. Thus dqsrc should be small enough (less than 1 degree) when jdefsrc=0.
- For specular reflection (DSM model), the method to determine a random reflection direction is not very accurate.

[Edit](#)

## MCARaTS v0.9: Sep. 14, 2006

1. The software package was renamed, and the version number was reset. Now released under GNU GPL (general public license) version 2.

2. Revised surface reflection model

- Pixel-by-pixel surface reflection can be now selected from black, Lambertian, DSM, and RPV models (see below).
  - Added Rahman-Pinty-Verstraete (RPV) BRDF model
  - Improved treatment of diffuse-specular mixture (DSM) reflection model
  - Thermal emission from non-Lambertian surface is now properly modeled
- From this modeling experience, we realized general treatment of BRDF models. Albedo is computed using LUT interpolation. Reflection direction is determined using a rejection method.

3. Air mass factor (AMF), which can be used for differential optical absorption spectroscopy (DOAS) techniques

- Can be computed for each pixel radiance. In addition, so-called "box AMF" can be computed for each layer.
- Effect of atmospheric sphericity (i.e. planet surface's curvature) is corrected for the line of sight (the last scattering point to the radiance detector).

4. New localized radiation source, adding Lidar-like simulation functionality.

- Radiation sources can be from a point. The source is assumed as isotropically emitting from a conical angular region. The center direction of the cone, the cone angle, emitting energy, the location, and geometrical size of the emitting sphere are specified by the user.

5. New localized plane flux or actinic flux (spheradiance)

- Can be computed for arbitrary inclined plane and for arbitrary conical field-of-view (FOV).
- 2-D distribution with respect to binned pathlengths and FOVs are obtained from one simulation. The pathlength is here geometrical pathlength from the source emission to the detector.

6. Added atmospheric refraction: Transmission and reflection at each layer interface are fully modeled.

- Refractive indexes are specified for each layer. Horizontal inhomogeneity of the indexes is neglected.

7. Improved variance reduction techniques

- Added numerical diffusion techniques, which improve the accuracy of radiative transfer simulation without significant increase of

computation time. The diffusion was implemented to fluxes, heating rates, radiances, and camera images, respectively. The computation efficiency (Monte Carlo noise vs CPU time) is improved by a factor of 2 to 100, depending on cases.

- Added collision-forcing method for optically-thin media. The delta scaling increases extinction coefficient. This dramatically improves accuracy of heating rates at optically-thin regions and radiances from optically-thin regions.
- Improved the truncation approximation. Accuracy of radiance is improved by this.
- Modified local estimation method, decreasing sampling frequency in an unbiased way. This accelerates computations of local estimates (pixel-radiances, camera radiances, and local fluxes)
- Revised the Russian roulette method, adding a "weight window". Efficiency for extreme cases with optically thick, non-absorbing atmosphere and strongly absorbing atmosphere are improved.

8. IR cooling rates are calculated in an analytical way

9. Other minor improvements and fixes

- Modified example cases
- Rayleigh scattering phase function is now included in the code itself. Can be specified by the phase function ID as 0.
- Added presets of technical parameters, getting easy to optimize numerical efficiency. A preset ID ("jtech") can be selected in the input file.

[Edit](#)

## MCRS v3.4.1: May 12, 2005

---

1. Minor updates for compatibility with some specific compilers.

[Edit](#)

## MCRS v3.4: Dec. 20, 2004

---

1. Infrared modeling including inertial emission

- The codes can now be applied to ultra-violet to microwave wavelength regions.
- Lambertian surface is assumed for the longwave regions in the current implementation.

2. A new hybrid 3-D RT solver, P3D (Partially-3-D) scheme, that uses ICA for higher-order scattering (the critical order is specified by the user)

3. Accuracy improvements and acceleration

- Incorporating a maximum cross section method for transfer in super-grids
- Accurate, scalable and flexible truncation method according to the order of scattering
- Scaling approximation applying isotropic scattering to higher-order scattering
- Scaling technique to treat gaseous absorption
- The Russian roulette method
- Experimental implementation of the photon splitting technique for variance reduction
- Experimental implementation of the local estimation method that can be optionally used for computations of fluxes and heating rates.

- MPI parallelization for the core part of simulation are modified. The horizontal domain are splitted in many subdomains, and each processor unit simulates trajectories that have one unique initial subdomain.

4. Flexible and realistic models for gases, aerosols, hydrometeors, and land and water surfaces

- Correlated k-distribution method for gaseous absorption. Mixing ratio of absorbing gases can vary three-dimensionally.
- Several media defined in 3-D grids (gases, aerosols, cloud water, cloud ice, rain, snow, graupel etc)
- Two media defined in 1-D grids
- A mixed bi-directional reflectance distribution function (BRDF) model of diffusive and specular reflectors for a realistic expression of surface: The diffusive reflection is modeled by a modified Lommel-Seeliger model. The specular (Fresnellian) reflection is modeled by a rough micro-facet model. Their fractions, minimum diffusive reflectance, asymmetry of the diffusive reflection, complex refractive index of the media, roughness of the micro-facets can vary spatially.
- Source (solar) disk size (angle) can be configured, while it was restricted as null in previous versions.
- The phase functions can be specified as Heyney-Greenstein function. The user inputs asymmetry factor for such a case.
- The solar azimuth angle can be random.
- Radiances can be computed at arbitrary altitudes though they are restricted to TOA/BOA in the previous versions. Direct beam radiance is added to diffuse radiance in the beam incoming direction.

5. Simulated camera images

- Camera positions and directions and field-of-views of the images are arbitrarily configured.
- Scaling method is used to suppress CPU time for oblique view image radiances.

## 6. Making the codes easier to use

Namelist input and simplified user interfaces. Large data arrays can be input from external input files. 3-D atmospheric data can

- be input from a text or binary file.
- Accuracy can be improved by using additional simulations with precomputed results

## 7. Revised pre/post-processing tools

- Wrote a pre-processing shell script, jobmodel.csh, to prepare input files for the RT simulation.
- Interfaces for visualizing input/output data for the mcrcs codes with GrADS.
- Reconstructed post-processing tools. The tool, paddnoise, was integrated in pconvert.

## 8. Other minor improvements and fixes

- Scattering angle is interpolated from look-up-table (LUT). The accuracy increased by this even when number of grids of the LUT are small.
- Input variables for mcrcs codes are checked before simulations.
- Example cases were modified.
- Some mathematical functions, such as  $\exp(-x)$  and  $-\ln(x)$ , are computed using LUTs for quick computation.
- Several initialization processes in mcrcs code are parallelized by MPI.
- When the number of incident photons are not integer, the incidence of the last photon are determined by a random number.

[Edit](#)

## MCRS v3.3: Dec. 29, 2003

---

### Major changes:

- The RT solver code is parallelized by MPI (Message Passing Interface). Parallelized code is generated by an option when compiling the codes (see ./src/Makefile).
- Fluxes are now computed every or several layer interfaces. The user can choose one setup from no flux, TOA and BOA, every 1D-layers, or every 1D and 3D-layers.
- Broken cloud modeling with several useful tools to generate artificial cloud realizations.

### Minor changes:

- RT solver code is accelerated by 5-30% from version 3.2.2 due to several minor improvements.
- More accurate algorithm for determining scattering angle with interpolation is used.
- Angular-averaged radiances are no longer computed.
- Several other minor improvements and fixes.

[Edit](#)

## MCRS v3.2.2: Dec. 26, 2003

---

### Minor changes:

- A minor bug in delta-scaling of the extinction coefficient according to the truncation of phase function was fixed (in mcrstrace?.d.F). By this, accuracy is improved even when the truncated delta fraction is very large.
- A better algorithm to treat phase functions and scattering angles with high accuracy is used.
- Several other very minor improvements and fixes.

[Edit](#)

## MCRS v3.2.1: Mar. 3, 2003

---

### Minor changes:

- Computational speed was improved by 20 (case5) to 70% (case4). This is achieved by changes in subroutines for photon tracing ("src/mcsrc/mcsrcbound3d.f" etc).
- Bugs in "src/model/mdlslice.f" and sample case-5 input data file were fixed.
- Documentation about "paddnoise" was added.

[Edit](#)

## MCRS v3.2: Jan. 18, 2003

---

### Major changes:

- An ability to compute radiance was added (it differs from angular-averaged radiance). Two variance reduction techniques were incorporated for radiance computation: a method of local estimate and truncation of forward peak of phase function. The truncation may be also effective for fast flux computation. The maximum truncation fraction is specified by user. Note the truncation fraction determines the absolute accuracy and computing efficiency. If the user needs an absolute accuracy, he/she

should use a large number of photons with small truncation fraction.

- The surface reflection algorithm was changed to the weighting method. By that, the accuracy with the same CPU time was improved.
- Plane-parallel approximation (PPA) was added as an option.
- Cloud modeling tools were added (see ./src/cloud).
- A direct flux computation was added.
- We changed the sample cases. Cases 4 and 5 are provided for I3RC phase II comparisons.
- More documentations.
- The adaptive gridding along z-axis was removed for simplicity.
- We removed the maximum cross section technique because gaseous absorption calculation based on pathlength is not accurate using the method. This removal is for simplicity of near-future development. In the future, the method might be added again.

Minor changes:

- A use of look-up table of  $\text{acos}()$  was added.
- The input data were separated into three files. The output data from mcrs are put to a unified file with four parts.
- LSIPA computation was removed for simplicity.

[Edit](#)

## MCRS v3.1.1: Sep. 26, 2002

---

Minor changes:

- A few bugs were fixed in mcrstrace3d0mcs.F & mcrstrace3d2.F. In the previous version, MCS & IPA calculations are not valid.

[Edit](#)

## MCRS v3.1: Sep. 20, 2002

---

Major changes:

- First public release. Manual documents were written. Compatibility and performances were tested on various machines.
- IPA (Independent Pixel Approximation) and LSIPA (Low-order scattering-IPA) calculations were introduced as the 3DRT approximations. In the public version, LSIPA is not included because it is under experiment.
- The maximum cross section (MCS) technique was incorporated.
- The user can give 3D arrangement of single scattering properties arbitrarily.
- Some utility tools to manipulate input/output data of mcrs were made.

[Edit](#)

---

---