

RADTRAD Change Log: Version 5.0.0

Released February 2020

- RADTRAD 4.x was converted from Fortran to java. While java is a modern language, many of the paradigms of Fortran were maintained in the code. RADTRAD 5.0.0 involved significant changes to modernize the structure of the code, remove limitations, and make the code more maintainable.
- There is no longer a limit on the number of components allowed or size of tables in a RADTRAD model.
- The code has been optimized to make the adaptive time stepping algorithm much faster.
- In previous versions, the adaptive time stepping algorithm calculated error only at the dose locations. Because of this, there was often a time delay between the event that caused the error (e.g., a gap release to containment) and the algorithm's measurement of that error (e.g., in a dose location many compartments downstream of containment). In such a scenario, the algorithm would attempt to decrease time steps to correct for a mechanism that had already passed, as it occurred many time steps before the error was measured. In order to address this, RADTRAD 5.0.0 calculates phantom doses in each compartment (doses that are not used or output in the formal calculation) and estimates error using these doses. This allows for error to be measured closer to the point of occurrence, improving the algorithms ability to adapt the time step accordingly.
- In addition to the L_2 (global) error norm, and L_∞ (local) error norm was added. The L_2 norm minimizes error in nuclides that contribute the most to dose. The L_∞ seeks to minimize error in all nuclides whether they contribute significantly to dose or not. The L_∞ error norm is expected to cause more timesteps to occur in general. However, in the RADTRAD 5.0.0 release, there is an implementation error in the L_∞ algorithm, and the L_2 norm is recommended. This is fixed in RADTRAD 5.0.1.
- The command line option '-M' or '--useMAXerror' was added, which causes the calculated error to be based on the maximum error over all dose values rather than using an averaged error that is weighted toward nuclides with larger dose concentrations.
- The 'r' or '--rgtest' (regression testing) command line option causes the plot and output file to remove markers that might differ in different versions of the code to simplify comparison. In RADTRAD 4.5.x, this option did not cause the cpu time plot variable to be removed from the plot file, so the command line option '-T' or '--cpu_time_off' had to be used as well for regression tests. In RADTRAD 5.0.0, the cpu time variable is removed when the '--rgtest' command line option is used.
- Legacy code to process RADTRAD 3.x input files was removed.

- The command line option '--addNucliPlots' was added. This causes the nuclide quantity in each compartment to be included in the plot file.
- The command line option '--addTransPlots' was added. This causes the quantity in each compartment of each transport group to be included in the plot file.

- **Code Fixes:**

- SNAP allowed RADTRAD component numbers to be specified manually. However, RADTRAD would fail or get incorrect results if the components were not ordered sequentially. This was corrected so that the user can customize the component numbering and get correct results.
- The adaptive time step algorithm would sometimes get into cycles where the timestep is doubled and then cut in half repeatedly, suggesting that the error calculation was not stable. Code was added that causes RADTRAD to wait 10 time steps after the time step size has been cut to increase the time step in order to improve stability.